



PCT/FR 2004 / 003009

14 DEC. 2004

REC'D 07 FEB 2005

WIPO

PCT

# BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

## COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le 22 NOV. 2004

Pour le Directeur général de l'Institut  
national de la propriété industrielle  
Le Chef du Département des brevets

Martine PLANCHE

DOCUMENT DE  
PRIORITÉ

PRÉSENTÉ OU TRANSMIS  
CONFORMÉMENT À LA RÈGLE  
17.1. a) OU b)

BEST AVAILABLE COPY

INSTITUT  
NATIONAL DE  
LA PROPRIÉTÉ  
INDUSTRIELLE

SIEGE  
26 bis, rue de Saint-Petersbourg  
75800 PARIS cedex 08  
Téléphone : 33 (0)1 53 04 53 04  
Télécopie : 33 (0)1 53 04 45 23  
www.inpi.fr

ETABLISSEMENT PUBLIC NATIONAL

CRÉE PAR LA LOI N° 51-444 DU 19 AVRIL 1951



26 bis, rue de Saint Pétersbourg  
75800 Paris Cedex 08  
Téléphone : 33 (1) 53 04 53 04 Télécopie : 33 (1) 42 94 86 54

# BREVET D'INVENTION CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa  
N° 11354\*02

## REQUÊTE EN DÉLIVRANCE page 1/2

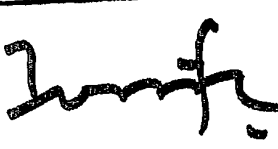
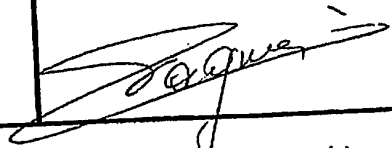
BR1

Cet imprimé est à remplir lisiblement à l'encre noire

08 540 019 / 010901

<b>REMISE DES PIÈCES</b> DATE <b>10 DEC 2003</b> LIEU <b>75 INPI PARIS 34 SP</b> N° D'ENREGISTREMENT <b>0314490</b> NATIONAL ATTRIBUÉ PAR L'INPI DATE DE DÉPÔT ATTRIBUÉE <b>10 DEC. 2003</b> PAR L'INPI		<b>1</b> NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE  CABINET PLASSERAUD 65/67 rue de la Victoire 75009 PARIS	
Vos références pour ce dossier (facultatif) AH-EMA/BFF030328			
Confirmation d'un dépôt par télécopie <input type="checkbox"/>		<input type="checkbox"/> N° attribué par l'INPI à la télécopie	
<b>2</b> NATURE DE LA DEMANDE		Cochez l'une des 4 cases suivantes	
Demande de brevet <input checked="" type="checkbox"/>			
Demande de certificat d'utilité <input type="checkbox"/>			
Demande divisionnaire <input type="checkbox"/>			
Demande de brevet initiale N° _____ Date _____			
ou demande de certificat d'utilité initiale N° _____ Date _____			
Transformation d'une demande de brevet européen <input type="checkbox"/>			
Demande de brevet initiale N° _____ Date _____			
<b>3</b> TITRE DE L'INVENTION (200 caractères ou espaces maximum)  PROCEDE DE CODAGE MULTIPLE OPTIMISE.			
<b>4</b> DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE FRANÇAISE		Pays ou organisation _____ N° _____ Date _____ Pays ou organisation _____ N° _____ Date _____ Pays ou organisation _____ N° _____ <input type="checkbox"/> S'il y a d'autres priorités, cochez la case et utilisez l'imprimé «Suite»	
<b>5</b> DEMANDEUR (Cochez l'une des 2 cases)		<input checked="" type="checkbox"/> Personne morale <input type="checkbox"/> Personne physique	
Nom ou dénomination sociale		FRANCE TELECOM	
Prénoms			
Forme juridique		Société Anonyme	
N° SIREN		3 8 0 1 2 9 8 6 6	
Code APE-NAF		1 1 1 1	
Domicile ou siège	Rue	6, Place d'Alleray	
	Code postal et ville	7 5 0 1 5 PARIS	
	Pays	FRANCE	
Nationalité		Française	
N° de téléphone (facultatif)		N° de télécopie (facultatif)	
Adresse électronique (facultatif)			
<input type="checkbox"/> S'il y a plus d'un demandeur, cochez la case et utilisez l'imprimé «Suite»			

Remplir impérativement la 2<sup>ème</sup> page

REMISE DES PIÈCES DATE <b>10 DEC 2003</b> LIEU <b>75 INPI PARIS 34 SP</b> N° D'ENREGISTREMENT <b>0314490</b> NATIONAL ATTRIBUÉ PAR L'INPI		Réservé à l'INPI	
<b>Vos références pour ce dossier :</b> <i>(facultatif)</i>		AH/EMA-BFF030328	
<b>6 MANDATAIRE</b> <i>(s'il y a lieu)</i>			
Nom			
Prénom			
Cabinet ou Société		CABINET PLASSERAUD	
N° de pouvoir permanent et/ou de lien contractuel			
Adresse	Rue	65/67 rue de la Victoire	
	Code postal et ville	75 009 PARIS	
	Pays	FRANCE	
N° de téléphone <i>(facultatif)</i>			
N° de télécopie <i>(facultatif)</i>			
Adresse électronique <i>(facultatif)</i>			
<b>7 INVENTEUR (S)</b>		Les inventeurs sont nécessairement des personnes physiques <input type="checkbox"/> Oui <input checked="" type="checkbox"/> Non : Dans ce cas remplir le formulaire de Désignation d'inventeur(s)	
Les demandeurs et les inventeurs sont les mêmes personnes		Uniquement pour une demande de brevet (y compris division et transformation)	
<b>8 RAPPORT DE RECHERCHE</b>		<input checked="" type="checkbox"/> Établissement immédiat <input type="checkbox"/> Établissement différé	
Paiement échelonné de la redevance <i>(en deux versements)</i>		Uniquement pour les personnes physiques effectuant elles-mêmes leur propre dépôt <input type="checkbox"/> Oui <input checked="" type="checkbox"/> Non	
<b>9 RÉDUCTION DU TAUX DES REDEVANCES</b>		Uniquement pour les personnes physiques <input type="checkbox"/> Requête pour la première fois pour cette invention <i>(joindre un avis de non-imposition)</i> <input type="checkbox"/> Obtenue antérieurement à ce dépôt pour cette invention <i>(joindre une copie de la décision d'admission à l'assistance gratuite ou indiquer sa référence)</i> : AG <input type="text"/>	
Si vous avez utilisé l'imprimé «Suite», indiquez le nombre de pages jointes			
<b>10 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE</b> (Nom et qualité du signataire) Raphaël LOUISET (CPI n° 02-1002)		VISA DE LA PRÉFECTURE OU DE L'INPI  	

Procédé de codage multiple optimisé

La présente invention concerne le codage/décodage de signaux numériques, dans des applications de transmission ou de stockage de signaux multimédia tels que les signaux audio (parole et/ou sons) ou vidéo.

Pour offrir mobilité et continuité, les services de communication multimédia modernes et innovants doivent pouvoir fonctionner dans une grande variété de conditions. Le dynamisme du secteur de la communication multimédia, l'hétérogénéité des réseaux, de l'accès et des terminaux ont engendré une prolifération de formats de compression.

La présente invention s'inscrit dans le contexte d'une optimisation des techniques de "codage multiple", mises en œuvre dès lors qu'un signal numérique, ou une portion de ce signal, est codé selon plusieurs techniques de codage. Ce codage multiple peut être effectué de manière simultanée (en une seule passe) ou non. Les traitements peuvent s'effectuer sur le même signal, ou éventuellement sur des versions dérivées du même signal (par exemple selon des bandes passantes différentes). On distingue donc le "codage multiple" des "transcodages", où chaque codeur effectue la compression d'une version issue du décodage du signal compressé par le codeur précédent.

Le codage multiple se présente par exemple dans le cas d'un même contenu qui est codé selon plusieurs formats et transmis ensuite à des terminaux ne supportant pas les mêmes formats de codage. S'il s'agit d'une diffusion en

temps réel, le traitement devra être effectué en simultané. S'il s'agit d'accès à une base de données, les codages pourront être effectués l'un après l'autre, en différé. Dans ces exemples, le codage multiple permet de  
5 coder un même signal selon des formats différents en utilisant plusieurs codeurs (ou éventuellement plusieurs débits ou plusieurs modes d'un même codeur), chaque codeur fonctionnant de manière indépendante des autres codeurs.

10 Un autre usage de codage multiple se rencontre dans des structures de codage où plusieurs codeurs se trouvent en compétition pour coder un segment de signal, un seul codeur étant finalement sélectionné pour le codage de ce segment. Le choix du codeur sélectionné peut s'effectuer à  
15 l'issue du traitement de ce segment, voire même ultérieurement (par décision retardée). Dans ce qui suit, on désignera par "codage multi-modes" ce type de structure (en référence à la sélection d'un "mode" de codage). Dans ces structures multi-modes, plusieurs codeurs partageant  
20 un "passé commun" sont amenés à coder la même portion de signal. Les techniques de codage utilisées peuvent être différentes, ou issues d'une unique structure de codage. Elles ne seront cependant pas totalement indépendantes, sauf s'il s'agit de techniques "sans mémoire". En effet,  
25 dans le cas (courant) de techniques de codage mettant en œuvre des traitements récursifs, le traitement d'un segment donné de signal dépend de la manière dont ce signal a été codé dans le passé. Il y a donc une certaine dépendance entre les codeurs, dès lors qu'un codeur devra  
30 prendre en compte dans ses mémoires la sortie d'un autre codeur.

Dans ces différents contextes, la notion de "codage multiple" a été introduite ainsi que les conditions d'usage de telles techniques. Cependant la complexité de  
5 mise en œuvre peut s'avérer rédhibitoire.

Par exemple, dans le cas de serveurs de contenus qui diffusent un même contenu sous plusieurs formats adaptés aux conditions d'accès, de réseaux et terminaux de  
10 différents clients, cette opération devient extrêmement complexe à mesure qu'augmente le nombre de formats désiré. S'il s'agit d'une diffusion temps réel, on se trouve rapidement limité par les ressources du système étant donné que les différents formats sont codés en parallèle.

15 Le deuxième cas d'usage mentionné concerne les applications de codage multi-modes, permettant la sélection d'un codeur parmi un ensemble pour chaque portion de signal analysé. La sélection demande la  
20 définition d'un critère, les plus courants visant à l'optimisation du compromis débit-distorsion. Le signal étant analysé sur des segments temporels successifs, à chaque segment plusieurs codages sont évalués. On sélectionne ensuite le codage de débit le plus faible pour  
25 une qualité donnée, ou celui de meilleure qualité pour un débit donné. On notera que d'autres contraintes que celles de débit/distorsion peuvent être utilisées.

En général, dans de telles structures, la sélection du  
30 codage s'effectue "a priori" par une analyse du signal sur le segment considéré (sélection selon les caractéristiques

du signal). Cependant, la difficulté de produire une classification robuste du signal pour cette sélection a conduit à proposer une sélection "a posteriori" du mode optimal après codage de l'ensemble des modes, au prix toutefois d'une complexité élevée.

Des méthodes intermédiaires combinant les deux approches ont été proposées pour alléger le coût de calcul. Ces stratégies sont cependant sous-optimales et s'avèrent moins performantes que l'exploration de tous les modes. L'exploration de tous les modes ou d'une grande partie des modes constitue une application de codage multiple qui présente une complexité potentiellement élevée, difficilement compatible a priori avec le codage en temps réel par exemple.

Actuellement, la plupart des opérations de codage multiple et de transcodage ne prennent pas en compte les interactions entre les formats et entre le format et son contenu. Quelques techniques de codage multi-modes ont été proposées, mais la décision du mode utilisé se fait généralement a priori soit sur le signal (par classification, comme par exemple le codeur SMV pour "Selectable Mode Vocoder"), soit en fonction des conditions du réseau (par exemple dans les codeurs AMR pour "Adaptive Multi-Rate").

Dans les documents :  
 "An overview of variable rate speech coding for cellular networks", Gersho, A.; Paksoy, E.; Wireless Communications, 1992. Conference Proceedings, 1992 IEEE

International Conference on Selected Topics, 25-26 Jun  
1992 Page(s): 172 -175,

5 "A variable rate speech coding algorithm for cellular  
networks", Paksoy, E.; Gersho, A.; Speech Coding for  
Telecommunications, 1993. Proceedings., IEEE Workshop  
1993, Page(s): 109-110,

10 "Variable rate speech coding for multiple access wireless  
networks", Paksoy E.; Gersho A.; Electrotechnical  
Conference, 1994, Proceedings, 7th Mediterranean, 12-14  
Apr 1994 Page(s): 47 -50 vol.1,

15 plusieurs modes de sélection sont présentés, en  
particulier une décision contrôlée par la source et une  
décision contrôlée par le réseau.

20 Dans le cas d'une décision contrôlée par la source, la  
décision a priori s'effectue à partir d'une classification  
du signal d'entrée. Il existe alors de nombreuses méthodes  
de classification du signal.

25 Dans le cas d'une décision contrôlée par le réseau, il est  
plus simple de réaliser un codeur multi-modes dont le  
débit est choisi par un module externe plutôt que par la  
source. La méthode la plus simple consiste à élaborer une  
famille de codeurs chacun à débit fixe mais dont les  
débits sont différents entre codeurs et de commuter entre  
ces différents débits pour obtenir un mode courant désiré.



Quelques travaux ont aussi été présentés sur la possibilité de combiner plusieurs critères pour sélectionner a priori le mode qui doit être utilisé, notamment dans les documents :

5 "Variable-rate for the basic speech service in UMTS" Berruto, E.; Sereno, D.; Vehicular Technology Conference, 1993 IEEE 43rd, 18-20 May 1993 Page(s): 520 -523

10 "A VR-CELP codec implementation for CDMA mobile communications" Cellario, L.; Sereno, D.; Giani, M.; Blocher, P.; Hellwig, K.; Acoustics, Speech, and Signal Processing, 1994, ICASSP-94, 1994 IEEE International Conference, Volume: 1 , 19-22 Apr 1994 Page(s): I/281 - I/284 vol.1.

15 Tous les algorithmes de codage multi-modes avec sélection du mode de codage a priori souffrent d'un même inconvénient, en particulier lié à des problèmes de robustesse de la classification a priori.

20 C'est pourquoi des techniques utilisant une décision a posteriori du mode de codage ont été proposées. Par exemple dans le document :

25 "Finite state CELP for variable rate speech coding" Vaseghi, S.V.; Acoustics, Speech, and Signal Processing, 1990, ICASSP-90, 1990 International Conference, 3-6 Apr 1990 Page(s): 37 -40 vol.1,

30 le codeur peut commuter entre différents modes par optimisation d'une mesure de qualité objective, la

décision se fait donc a posteriori en fonction des caractéristiques du signal d'entrée, du rapport visé débit/SQNR (pour "Signal to Quantization Noise Ratio") et de l'état courant du codeur. Un tel schéma de codage permet d'obtenir une amélioration de la qualité. Cependant, les différents codages étant réalisés en parallèle, la complexité résultante de ce type de système est prohibitive.

D'autres techniques combinant une décision a priori et une amélioration en boucle fermée ont été proposées. Dans le document :

"Multimode variable bit rate speech coding: an efficient paradigm for high-quality low-rate representation of speech signal" Das, A.; DeJaco, A.; Manjunath, S.; Ananthapadmanabhan, A.; Huang, J.; Choy, E.; Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings, 1999 IEEE International Conference, Volume: 4, 15-19 Mar 1999 Page(s): 2307 -2310 vol.4,

le système proposé effectue une première sélection (sélection en boucle ouverte) du mode, en fonction des caractéristiques du signal. Cette décision peut être effectuée par classification. Ensuite, à partir d'une mesure d'erreur, si les performances du mode sélectionné ne sont pas satisfaisantes, un mode de débit plus élevé est appliqué et l'opération se répète (selon une décision recherchée en boucle fermée).

30

De même, dans les documents :

\* "Variable rate speech coding for umts" Cellario, L.; Sereno, D.; Speech Coding for Telecommunications, 1993. Proceedings, IEEE Workshop, 1993 Page(s): 1 -2

5 "Phonetically-based vector excitation coding of speech at 3.6 kbps" Wang, S.; Gersho, A.; Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference, 23-26 May 1989 Page(s): 49 -52 vol.1

10 \* "A modified CS-ACELP algorithm for variable-rate speech coding robust in noisy environments" Beritelli, F.; IEEE Signal Processing Letters, Volume: 6 Issue: 2, Feb 1999 Page(s): 31 -34,

15 des techniques similaires ont été utilisées. Une première sélection en boucle ouverte est réalisée après classification du signal d'entrée (classification phonétique, ou voisé/non-voisé), ensuite une décision en boucle fermée est effectuée :

- 20     - soit sur le codeur complet et, dans ce cas, tout le segment de parole est codé à nouveau;
- soit sur une partie du codage, comme dans les références ci-avant précédées d'une étoile (\*), pour lesquels le choix du dictionnaire à utiliser est
- 25     effectué en boucle fermée.

L'ensemble des études mentionnées ci-dessus tend à résoudre le problème de la complexité de la sélection optimale du mode par l'utilisation, totale ou partielle,

30 d'une sélection ou pré-sélection a priori, qui évite le

codage multiple ou diminue le nombre de codeurs à mettre en œuvre en parallèle.

5 Toutefois, aucune technique de l'art antérieur permettant de réduire la complexité des codages réalisés en parallèle n'a été proposée.

La présente invention vient améliorer la situation.

10 Elle propose à cet effet un procédé de codage multiple en compression, dans lequel un signal d'entrée est destiné à alimenter en parallèle une pluralité de codeurs comportant chacun une succession de blocs fonctionnels, en vue d'un codage en compression dudit signal par chaque codeur.

15

Le procédé de l'invention comporte les étapes préparatoires ci-après :

- a) identifier les blocs fonctionnels formant chaque codeur, ainsi qu'une ou plusieurs fonctions réalisées par
- 20 chaque bloc,
- b) repérer, parmi lesdites fonctions, des fonctions qui sont communes d'un codeur à l'autre, et
- c) exécuter lesdites fonctions communes, une fois pour toutes, pour une partie au moins de tous les codeurs, au
- 25 sein d'au moins un même module de calcul.

Dans une réalisation avantageuse, les étapes ci-avant sont mises en œuvre par un produit programme d'ordinateur comportant des instructions de programme à cet effet. A ce

30 titre, la présente invention vise aussi un tel produit programme d'ordinateur, destiné à être stocké dans une

mémoire d'une unité de traitement, notamment d'un ordinateur ou d'un terminal mobile, ou sur un support mémoire amovible et destiné à coopérer avec un lecteur de l'unité de traitement.

5

La présente invention vise aussi un dispositif d'aide à un codage en compression, pour la mise en œuvre du procédé selon l'invention, et comportant alors une mémoire propre à stocker des instructions d'un produit programme d'ordinateur du type précité.

10

D'autres caractéristiques et avantages de l'invention apparaîtront à l'examen de la description détaillée ci-après, et des dessins annexés sur lesquels :

- 15 - la figure 1a illustre schématiquement le contexte d'application de la présente invention, avec une pluralité de codeurs mis en parallèle,
- la figure 1b illustre schématiquement l'application de l'invention, avec la mise en partage de blocs fonctionnels entre plusieurs codeurs mis en parallèle,
- 20 - la figure 1c illustre schématiquement l'application de l'invention, avec la mise en partage de blocs fonctionnels en codage multi-modes,
- la figure 1d illustre schématiquement l'application de l'invention, en codage multi-modes en treillis,
- 25 - la figure 2 représente schématiquement les blocs fonctionnels principaux d'un codeur fréquentiel perceptuel,
- la figure 3 représente schématiquement les blocs fonctionnels principaux d'un codeur à analyse par
- 30 synthèse,

- la figure 4a représente schématiquement les blocs fonctionnels principaux d'un codeur TDAC,
- la figure 4b représente schématiquement le format du flux binaire codé par le codeur de la figure 4a,
- 5 - la figure 5 représente schématiquement l'application de l'invention à une pluralité de codeurs TDAC en parallèle, selon une réalisation avantageuse,
- la figure 6a représente schématiquement les blocs fonctionnels principaux d'un codeur MPEG-1 (layer I et
- 10 II),
- la figure 6b représente schématiquement le format du flux binaire codé par le codeur de la figure 6a,
- la figure 7 représente schématiquement l'application de l'invention à une pluralité de codeurs MPEG-1 (layer I et
- 15 II) mis en parallèle, selon une réalisation avantageuse,
- et la figure 8 représente plus en détails les blocs fonctionnels d'un codeur à analyse par synthèse, ici de type NB-AMR selon la norme 3GPP.

20 On se réfère tout d'abord à la figure 1a sur laquelle on a représenté une pluralité de codeurs C0, C1, ..., CN, en parallèle et recevant chacun un signal d'entrée  $s_0$ . Chaque codeur comporte des blocs fonctionnels BF1 à BF<sub>n</sub> pour mettre en œuvre des étapes successives de codage et

25 délivrer finalement un flux binaire codé BS0, BS1, ..., BSN. On indique en outre que dans une application en codage multi-modes, les sorties des codeurs C0 à CN sont reliées à un module MM de sélection du mode optimal et le flux binaire BS du codeur optimal est transmis (flèches en

30 traits pointillés de la figure 1a).

Pour une raison de simplicité, tous les codeurs de l'exemple de la figure 1a ont le même nombre de blocs fonctionnels, mais bien entendu tous ces blocs fonctionnels ne sont pas nécessairement prévus dans tous les codeurs, en pratique.

Certains blocs fonctionnels BFi sont parfois identiques d'un mode (ou d'un codeur) à l'autre, d'autres ne diffèrent qu'au niveau des quantificateurs. Des relations exploitables existent aussi lorsque l'on utilise des codeurs issus d'une même famille de codage, utilisant des modèles similaires ou calculant des paramètres liés physiquement au signal.

Ce sont ces relations que la présente invention propose d'exploiter, afin de réduire la complexité des opérations de codage multiple.

Dans un premier temps, l'invention propose d'identifier les blocs fonctionnels composant chacun des codeurs. On exploite ensuite les similarités techniques entre les codeurs en considérant les blocs fonctionnels dont les fonctions sont équivalentes ou voisines. Pour chacun de ces blocs, l'invention propose :

- d'une part de définir des opérations dites "communes", et de les effectuer une seule fois pour l'ensemble des codeurs;
- d'autre part, de mettre en œuvre des méthodes de calcul spécifiques à chaque codeur et utilisant notamment les résultats de ces calculs communs. Ces méthodes de calcul produisent un résultat

éventuellement différent de celui produit par un codage complet. L'objectif est alors en fait d'accélérer les traitements en exploitant les informations disponibles et fournies notamment par les calculs communs. De tels procédés permettant d'accélérer les calculs sont par exemple mis en œuvre dans des techniques destinées à réduire la complexité des opérations de transcodage (dites techniques de "transcodage intelligent").

10

La figure 1b illustre la solution proposée. Dans l'exemple représenté, les opérations "communes" précitées sont effectuées une seule fois pour une partie au moins des codeurs et, préférentiellement, pour l'ensemble des codeurs, dans un module indépendant MI qui redistribuera les résultats obtenus à une partie au moins des codeurs, ou préférentiellement à tous ces codeurs. Il s'agit ainsi d'une mise en partage entre une partie au moins de tous les codeurs C0 à CN (ou "mutualisation" ci-après) des résultats obtenus. Un tel module indépendant MI peut faire partie d'un dispositif d'aide à un codage multiple en compression tel que défini ci-avant.

Dans une variante avantageuse, plutôt que d'avoir recours à un module de calcul externe MI, on utilise le ou les blocs fonctionnels existants BF1 à BF<sub>n</sub> d'un même ou de plusieurs codeurs distincts, ce ou ces codeurs étant choisis selon des critères qui seront décrits plus loin.



La présente invention peut mettre en œuvre plusieurs stratégies qui, bien entendu, peuvent différer selon le rôle du bloc fonctionnel considéré.

5 Une première stratégie consiste à utiliser les paramètres du codeur dont le débit est le plus faible pour focaliser la recherche des paramètres pour tous les autres modes.

10 A l'inverse, une deuxième stratégie consiste à utiliser les paramètres du codeur dont le débit est le plus élevé, puis de "dégrader" progressivement jusqu'au codeur dont le débit est le plus faible.

15 Bien entendu, si l'on souhaite privilégier un codeur particulier, il est possible de coder un segment de signal en utilisant ce codeur, puis, en appliquant les deux stratégies ci-avant, d'atteindre les codeurs de débit supérieur et inférieur.

20 Bien entendu, d'autres critères que le débit peuvent être utilisés pour piloter la recherche. On peut par exemple, pour certains blocs fonctionnels, favoriser le codeur dont les paramètres se prêtent le mieux à une extraction (ou une analyse) et/ou à un codage efficaces des paramètres  
25 similaires des autres codeurs, l'efficacité pouvant être jugée selon la complexité, la qualité ou un compromis des deux.

Il peut être prévu aussi de créer un module de codage  
30 indépendant, non présent dans les codeurs, mais permettant

un codage plus efficace des paramètres du bloc fonctionnel considéré, pour l'ensemble des codeurs.

5 Ces diverses stratégies de mise en œuvre sont particulièrement intéressantes dans le cas du codage multi-modes. Dans ce contexte illustré à la figure 1c, la présente invention permet de réduire la complexité des calculs préliminaires à la sélection a posteriori d'un codeur effectuée en dernière étape, par exemple par le  
10 dernier module MM avant la transmission du flux binaire BS.

Dans ce cas particulier du codage multi-modes, une variante de la présente invention, représentée dans  
15 l'exemple de la figure 1c, propose d'introduire un module de sélection partielle  $MSPi$  (avec  $i = 1, 2, \dots, N$ ) après chaque étape de codage (donc après les blocs fonctionnels  $BFi1$  à  $BFiN_1$  mis en compétition et dont le résultat du ou des blocs sélectionnés  $BFicc$  va être utilisé par la  
20 suite). Ainsi, les similitudes entre les différents modes sont exploitées pour accélérer le calcul de chaque bloc fonctionnel. Tous les schémas de codage ne seront alors pas obligatoirement évalués.

25 Une variante plus sophistiquée de la structure multi-modes reposant sur la découpe en blocs fonctionnels décrite ci-avant est maintenant proposée, en référence à la figure 1d. La structure multi-modes de la figure 1d est dite "en treillis", avec plusieurs chemins possibles dans le  
30 treillis. En fait, sur la figure 1d, on a représenté tous les chemins possibles du treillis de sorte qu'il se

présente sous une forme arborescente. On indique en particulier que chaque chemin du treillis est défini par une combinaison de modes de fonctionnement des blocs fonctionnels, chaque bloc fonctionnel alimentant plusieurs variantes possibles du bloc fonctionnel suivant.

Ainsi, chaque mode de codage est issu de la combinaison de modes de fonctionnement des blocs fonctionnels : le bloc fonctionnel 1 possède  $N_1$  modes de fonctionnement, le bloc fonctionnel 2 en possède  $N_2$ , et ainsi de suite jusqu'au bloc P. L'ensemble des  $NN = N_1 \times N_2 \times \dots \times N_p$  combinaisons possibles est donc représenté par un treillis de  $NN$  branches décrivant, bout-à-bout, un codeur multi-modes complet à  $NN$  modes. On peut éventuellement supprimer a priori certaines branches du treillis et définir ainsi une arborescence comportant un nombre réduit de branches. Une première particularité de cette structure est qu'elle prévoit, pour un bloc fonctionnel donné, un module de calculs communs par sortie du bloc fonctionnel précédent. Ces modules de calculs communs effectuent les mêmes opérations, mais sur la base de signaux différents puisqu'ils sont issus de blocs antérieurs différents. Avantageusement, les modules de calculs communs d'un même niveau sont mutualisés : les résultats d'un module donné exploitables par les modules suivants sont fournis à ces modules suivants. D'autre part, une sélection partielle, effectuée à l'issue du traitement de chaque bloc fonctionnel, permet avantageusement de supprimer les branches les moins performantes selon le critère choisi. On peut donc réduire le nombre de branches du treillis à évaluer.

Une application avantageuse de cette structure multi-modes en treillis est la suivante.

5 Lorsque les blocs fonctionnels sont susceptibles d'opérer à des débits respectifs différents et en utilisant des paramètres respectifs propres auxdits débits, pour un bloc fonctionnel donné, le chemin du treillis choisi est celui traversant le bloc fonctionnel de débit le plus faible, ou encore le bloc fonctionnel de débit le plus élevé selon le  
10 contexte de codage, et les résultats obtenus du bloc fonctionnel de débit le plus faible (ou le plus élevé) sont adaptés aux débits d'une partie au moins des autres blocs fonctionnels par une recherche focalisée de paramètres pour une partie au moins de tous les autres  
15 blocs fonctionnels, jusqu'au bloc fonctionnel de débit le plus élevé (ou respectivement le plus faible).

En variante, on choisit un bloc fonctionnel de débit donné et on adapte progressivement au moins une partie des  
20 paramètres propres à ce bloc fonctionnel:  
- jusqu'au bloc fonctionnel capable d'opérer au débit le plus faible, par recherche focalisée, et  
- jusqu'au bloc fonctionnel capable d'opérer au débit le plus élevé, par recherche focalisée.

25 De manière générale, on réduit ainsi la complexité associée au codage multiple.

L'invention s'applique à tout schéma de compression  
30 mettant en œuvre le codage multiple d'un contenu

multimédia. Trois exemples de réalisation sont présentés dans ce qui suit, dans le domaine de la compression audio (parole et son). Les deux premiers exemples de réalisation se situent dans le contexte de la famille des codeurs par transformée, dont on peut donner le document suivant à

titre de référence :  
 "Perceptual Coding of Digital Audio", Painter, T.; Spanias, A.; Proceedings of the IEEE, Vol. 88, No 4, April 2000.

Le troisième exemple de réalisation se situe dans le contexte des codeurs CELP, dont on peut donner le document suivant à titre de référence :

"Code Excited Linear Prediction (CELP): High quality speech at very low bit rates" Schroeder M.R.; Atal B.S.; Acoustics, Speech, and Signal Processing, 1985. Proceedings. 1985 IEEE International Conference, Page(s): 937-940.

Un rappel des principales caractéristiques de ces deux familles de codage est tout d'abord présenté dans ce qui suit.

#### **\* Les codeurs par transformée ou en sous bandes**

Il s'agit de codeurs en compression par transformée ou en sous bandes basés sur des critères psychoacoustiques. Ce type de codeur procède par transformation sur des blocs du signal temporel pour obtenir un ensemble de coefficients. Les transformations sont du type temps-fréquence, l'une des transformations les plus utilisées étant la transformée en cosinus discrète modifiée (dit "MDCT", de

l'anglais "*Modified Discrete Cosine Transform*"). Avant la quantification de ces coefficients, un algorithme procède à l'allocation des bits de façon à ce que le bruit de quantification soit le moins audible possible.

5 L'allocation binaire et la quantification des coefficients mettent en œuvre une courbe de masquage, obtenue à l'aide d'un modèle psychoacoustique permettant d'évaluer, pour chaque raie spectrale considérée, un seuil de masquage représentatif de l'amplitude nécessaire pour qu'un son à

10 cette fréquence soit audible. La figure 2 donne le schéma de principe d'un codeur fréquentiel. On remarquera que la structure sous forme de blocs fonctionnels est bien représentée. En se référant à la figure 2, les blocs fonctionnels principaux sont :

- 15 - un bloc 21 de transformation temps/fréquence du signal audionumérique d'entrée  $s_0$ ,
- un bloc 22 de détermination d'un modèle perceptuel à partir du signal transformé,
- un bloc 23 de quantification et codage, à partir du
- 20 modèle perceptuel,
- et un bloc 24 de formatage du flux binaire pour obtenir une trame audio codée  $s_{tc}$ .

#### **\* Les codeurs à analyse par synthèse (codage CELP)**

25 Dans les codeurs à analyse par synthèse, le modèle de synthèse du signal reconstruit est utilisé au codeur pour extraire les paramètres modélisant les signaux à coder. Ces signaux peuvent être échantillonnés à la fréquence de 8 kHz (bande téléphonique 300-3400 Hz) ou à une fréquence

30 plus élevée, par exemple à 16 kHz pour le codage en bande élargie (bande passante de 50Hz à 7 kHz). Selon

l'application et la qualité désirée, le taux de compression varie de 1 à 16. Ces codeurs fonctionnent à des débits de 2 à 16 kbit/s en bande téléphonique, et à des débits de 6 à 32 kbit/s en bande élargie. Le dispositif de codage numérique de type CELP, codeur à analyse par synthèse le plus utilisé actuellement, est présenté à la figure 3 sous forme de blocs fonctionnels principaux. Le signal de parole  $s_0$  est échantillonné et converti en une suite de trames d'un nombre  $L$  d'échantillons. Chaque trame est synthétisée en filtrant une forme d'onde extraite d'un répertoire (appelé "dictionnaire"), multipliée par un gain, à travers deux filtres variant dans le temps. Le dictionnaire d'excitation fixe est un ensemble fini de formes d'ondes des  $L$  échantillons. Le premier filtre est un filtre de prédiction à long terme. Une analyse "LTP" (pour "Long Term Prediction") permet d'évaluer les paramètres de ce prédicteur à long terme qui exploite la périodicité des sons voisés, cette composante harmonique étant modélisée sous la forme d'un dictionnaire adaptatif (bloc 32). Le second filtre est un filtre de prédiction à court terme. Les méthodes d'analyse "LPC" (pour "Linear Prediction Coding") permettent d'obtenir ces paramètres de prédiction à court terme, représentatifs de la fonction de transfert du conduit vocal et caractéristiques de l'enveloppe du spectre du signal. Le procédé utilisé pour déterminer la séquence d'innovation est la méthode d'analyse par synthèse qui se résume comme suit. Au codeur, un grand nombre de séquences d'innovation du dictionnaire d'excitation fixes sont filtrées par le filtre LPC (filtre de synthèse du bloc fonctionnel 34 de la figure 3). Au

préalable, l'excitation adaptative a été obtenue de façon similaire. La forme d'onde sélectionnée est celle produisant le signal synthétique le plus proche du signal original (minimisation de l'erreur au niveau du bloc  
5 fonctionnel 35), selon un critère de pondération perceptuelle (bloc fonctionnel 36) qui est connu en général sous le nom de critère "CELP".

Dans le schéma de principe du codeur CELP donné à la  
10 figure 3, l'extraction de la fréquence fondamentale des sons voisés (ou "pitch"), appliquée sur le signal résultant de l'analyse LPC du bloc 31, permet ensuite d'en extraire la corrélation à long terme au niveau du bloc 32, appelée composante harmonique ou excitation adaptative  
15 (E.A.). Le signal résiduel est enfin modélisé classiquement par quelques impulsions, dont l'ensemble des positions est prédéfini dans un répertoire, appelé répertoire d'excitation fixe (E.F) dans le bloc 33.

20 Le décodage est, quant à lui, beaucoup moins complexe que le codage. Le flux binaire généré par le codeur permet au décodeur, après démultiplexage, d'obtenir l'index de quantification de chaque paramètre. Le décodage des paramètres et l'application du modèle de synthèse  
25 permettent alors de reconstruire le signal.

On décrit ci-après les trois exemples de réalisation précitées, en commençant tout d'abord par un codeur par transformée du type représenté sur la figure 2.



**\* Premier exemple de réalisation : application à un codeur  
"TDAC"**

Le premier exemple de réalisation concerne le codeur  
5 fréquentiel perceptuel dit "TDAC" et décrit notamment dans  
le document publié US-2001/027393. Ce codeur TDAC est  
utilisé pour coder des signaux audio numériques  
échantillonnés à 16 kHz (bande élargie). La figure 4a  
illustre les blocs fonctionnels principaux de ce codeur.  
10 Un signal audio  $x(n)$  limité en bande à 7 kHz et  
échantillonné à 16 kHz est découpé en trames de 320  
échantillons (20 ms). Une transformée en cosinus discrète  
modifiée (ou "MDCT") est appliquée (bloc fonctionnel 41)  
sur des trames du signal d'entrée de 640 échantillons avec  
15 un recouvrement de 50 %, donc avec un rafraîchissement de  
l'analyse MDCT toutes les 20 ms. On limite le spectre à  
7225 Hz en mettant à zéro les 31 derniers coefficients.  
(seuls les 289 premiers coefficients sont différents  
de 0). Une courbe de masquage (bloc 42) est déterminée à  
20 partir de ce spectre et tous les coefficients masqués sont  
mis à zéro. Le spectre est divisé en 32 bandes de largeurs  
inégaies. Les éventuelles bandes masquées sont déterminées  
en fonction des coefficients transformés des signaux. Pour  
chaque bande du spectre, l'énergie des coefficients MDCT  
25 est calculée (pour obtenir des facteurs d'échelle). Les 32  
facteurs d'échelle constituent l'enveloppe spectrale du  
signal qui est ensuite quantifiée puis codée par un codage  
entropique (bloc fonctionnel 43), et enfin transmise dans  
la trame codée  $s_c$ .

L'allocation dynamique des bits (bloc fonctionnel 44) se base sur une courbe de masquage par bande (bloc fonctionnel 42) calculée à partir de la version décodée et déquantifiée de l'enveloppe spectrale. Cette mesure permet  
 5 d'avoir une compatibilité entre l'allocation binaire du codeur et du décodeur. Les coefficients MDCT normalisés dans chaque bande sont ensuite quantifiés (bloc fonctionnel 45) par des quantificateurs vectoriels utilisant des dictionnaires imbriqués en taille, les  
 10 dictionnaires étant composés d'une union de codes à permutation de type II. Finalement, en se référant à la figure 4b, les informations sur la tonalité (codées ici sur un bit  $B_1$ ) et le voisement (codées ici sur un bit  $B_0$ ), ainsi que l'enveloppe spectrale  $e_q(i)$  et les coefficients  
 15 codés  $y_q(j)$  sont multiplexés (bloc 46 de la figure 4a) et transmis en trames.

Ce codeur pouvant fonctionner à plusieurs débits, on se propose de réaliser un codeur multi-débits par exemple à  
 20 16, 24 et 32 kbit/s. Dans ce schéma de codage, les blocs fonctionnels suivants peuvent être mis en commun entre les différents modes:

- Transformée MDCT (bloc 41),
- Détection de voisement (bloc fonctionnel 47 de la  
 25 figure 4a) et détection de tonalité (bloc fonctionnel 48 de la figure 4a),
- Calcul, quantification et codage entropique de l'enveloppe spectrale (bloc 43),

- Calcul d'une courbe de masquage, coefficient par coefficient, et d'une courbe de masquage par bande (bloc 42).

5 Ces différents blocs constituent 61,5% de la complexité du traitement dans le processus de codage. Leur factorisation est donc d'un intérêt important pour réduire cette complexité lors de la génération de plusieurs flux binaires correspondants à des débits différents.

10 Les résultats de ces blocs fonctionnels permettent déjà d'obtenir une première partie commune à tous les flux binaires de sortie qui contient les bits d'information sur le voisement, la tonalité et l'enveloppe spectrale codée.

15 Dans une première variante de cet exemple de réalisation, il est possible de réaliser les opérations d'allocation des bits et de quantification pour chacun des flux binaires de sortie correspondant à chacun des débits  
20 binaires considérés. Ces deux opérations sont effectuées exactement de la même manière qu'habituellement dans un codeur TDAC.

25 Dans une seconde variante plus avancée telle qu'illustrée sur la figure 5, on peut mettre en œuvre des techniques de transcodage "intelligent" (comme décrit dans le document publié US-2001/027393 cité ci-avant) pour réduire davantage la complexité et mutualiser certaines opérations, notamment :

- 30 - l'allocation de bits (bloc fonctionnel 44),

- et aussi la quantification des coefficients (blocs fonctionnels 45\_i), comme on le verra ci-après.

Sur la figure 5, les blocs fonctionnels mis en partage entre les codeurs (ou "mutualisés") portent la même  
 5 référence que ceux d'un seul codeur TDAC tel que représenté sur la figure 4a. Il s'agit des blocs 41, 42, 47, 48, 43 et 44. En particulier, le bloc 44 d'allocation des bits est utilisé en plusieurs passes, et le nombre de bits alloués est ajusté pour la transquantification  
 10 qu'effectue chaque codeur (blocs 45\_1, ... , 45\_(K-2), 45\_(K-1)), comme on le verra ci-après. On remarque en outre que ces transquantifications utilisent les résultats obtenus par le bloc fonctionnel 45\_0 de quantification pour un codeur choisi, d'indice 0 (le codeur de débit le  
 15 plus faible dans l'exemple décrit). Finalement, les seuls blocs fonctionnels des codeurs qui agissent sans interaction réelle sont les blocs de multiplexage 46\_0, 46\_1, ..., 46\_(K-2), 46\_(K-1), bien qu'ils utilisent tous les mêmes informations de voisement et de tonalité, ainsi que  
 20 la même enveloppe spectrale codée. A ce titre, on indique simplement qu'une mutualisation partielle du multiplexage peut être menée, là encore.

Pour les deux blocs fonctionnels d'allocation de bits et  
 25 de quantification, la stratégie employée consiste à exploiter les résultats des deux blocs fonctionnels d'allocation des bits et de quantification réalisés pour le flux binaire (0), au débit le plus bas  $D_0$ , pour accélérer les opérations des deux blocs fonctionnels  
 30 correspondants pour les  $K-1$  autres flux binaire ( $k$ ) ( $1 \leq k < K$ ). On peut aussi considérer le schéma de codage

multi-débits qui utilise un bloc fonctionnel d'allocation de bits par flux binaire (sans factorisation prévue pour ce bloc) mais mutualise une partie des opérations de quantification ensuite.

5

Les techniques de codage multiple présentées ci-après se basent avantageusement sur un transcodage intelligent utilisé pour la réduction du débit binaire de flux audio codé, généralement situé dans un nœud du réseau.

10

Dans la suite, les flux binaires  $k$ ,  $0 \leq k < K$ , sont classés suivant un ordre croissant de débits ( $D_0 < D_1 < \dots < D_{K-1}$ ). Ainsi, le flux binaire 0 correspond au débit binaire le plus bas.

15

#### \* Allocation de bits

L'allocation de bits dans le codeur TDAC se réalise en deux phases. D'abord un premier calcul du nombre de bits à allouer à chaque bande est effectué de préférence suivant

20 la formule suivante :

$$b_{opt}(i) = \frac{1}{2} \log_2 \left[ \frac{e_q^2(i)}{S_b(i)} \right] + C, \quad 0 \leq i \leq M-1,$$

où  $C = \frac{B}{M} - \frac{1}{2M} \sum_{l=0}^{M-1} \log_2 [e_q^2(l)/S_b(l)]$  est une constante,

$B$  est le nombre total de bits disponibles,

$M$  est le nombre de bandes,

25  $e_q(i)$  est la valeur décodée et déquantifiée de l'enveloppe spectrale sur la bande  $i$ ,

et  $S_b(i)$  est le seuil de masquage pour cette bande.

Chacune des valeurs obtenues est arrondie à l'entier naturel le plus proche. Si le débit total alloué n'est pas exactement égal à celui disponible, une seconde phase est utilisée pour réaliser le réajustement. Cette étape se fait préférentiellement par une succession d'opérations itératives basées sur un critère perceptuel qui ajoute ou retire des bits aux bandes.

Ainsi, si le nombre total de bits distribués est inférieur à celui disponible, l'ajout de bits se fait aux bandes où l'amélioration perceptuelle est la plus importante. Cette amélioration perceptuelle est mesurée par la variation du rapport bruit à masque entre l'allocation initiale et finale des bandes. Le débit est augmenté pour la bande où cette variation est la plus grande. Dans le cas contraire où le nombre total de bits distribués est supérieur à celui disponible, l'extraction de bits sur les bandes se fait de manière duale à cette dernière procédure.

Dans le schéma de codage multi-débits correspondant au codeur TDAC, il est possible de factoriser certaines opérations pour l'allocation de bits. Ainsi, la première phase de détermination par la formule ci-avant peut se faire une seule fois en se basant sur le débit binaire  $D_0$  le plus bas. La phase de ré-ajustement en ajoutant des bits peut se faire ensuite de manière continue. Une fois que le nombre total de bit distribué atteint le nombre correspondant à un débit binaire d'un flux binaire  $k$ ,  $k=1,2,\dots,K-1$ , la distribution courante est considérée comme celle qui est utilisée pour la quantification des vecteurs de coefficients normalisés par bande de ce flux binaire.

\* Quantification des coefficients

Pour ce qui concerne la quantification des coefficients, le codeur TDAC utilise une quantification vectorielle utilisant des dictionnaires imbriqués en taille, les dictionnaires étant composés d'une union de codes à permutation de type II. Ce type de quantification s'applique sur chacun des vecteurs des coefficients MDCT sur une bande. Un tel vecteur est normalisé au préalable en utilisant la valeur déquantifiée de l'enveloppe spectrale sur cette bande. On note :

- $C(b_i, d_i)$  le dictionnaire correspondant au nombre de bits  $b_i$  et à la dimension  $d_i$ ,
- $N(b_i, d_i)$  le nombre d'éléments dans ce dictionnaire,
- $CL(b_i, d_i)$  l'ensemble de ses leaders, et
- $NL(b_i, d_i)$  le nombre de leaders.

Le résultat de quantification pour chaque bande  $i$  de la trame est un mot de code  $m_i$  transmis dans le flux binaire.

Il représente l'index du vecteur quantifié dans le dictionnaire et calculé à partir des informations suivantes :

- le numéro  $L_i$ , dans l'ensemble  $CL(b_i, d_i)$  des leaders du dictionnaire  $C(b_i, d_i)$ , du vecteur leader quantifié  $\tilde{Y}_q(i)$  plus proche voisin d'un leader courant  $\tilde{Y}(i)$ ,
- le rang  $r_i$  de  $Y_q(i)$  dans la classe du leader  $\tilde{Y}_q(i)$ ,
- et la combinaison de signes  $sign_q(i)$  à appliquer à  $Y_q(i)$  (ou à  $\tilde{Y}_q(i)$ ),

où l'on précise les notations suivantes :

- $Y(i)$  est le vecteur des valeurs absolues des coefficients normalisés de la bande  $i$ ,
- $sign(i)$  est le vecteur des signes des coefficients normalisés de la bande  $i$ ,
- 5 •  $\tilde{Y}(i)$  est le vecteur leader du vecteur  $Y(i)$  précité, obtenu par ordonnancement décroissant de ses composantes (la permutation correspondante est notée  $perm(i)$ ),
- et  $Y_q(i)$  est le vecteur quantifié de  $Y(i)$  (ou "le plus  
10 proche voisin" de  $Y(i)$  dans le dictionnaire  $C(b_i, d_i)$ ).

Dans la suite, la notation  $\alpha^{(k)}$ , avec un exposant  $k$ , indique le paramètre utilisé dans le traitement effectué pour obtenir le flux binaire du codeur  $k$ . Les paramètres  
15 sans cet exposant étant calculés une seule fois pour toutes pour le flux binaire 0. Ils sont indépendants du débit (ou du mode) considéré.

La propriété "d'imbrication" des dictionnaires précitée  
20 s'exprime selon la relation :

$$C(b_i^{(0)}, d_i) \subseteq \dots \subseteq C(b_i^{(k-1)}, d_i) \subseteq C(b_i^{(k)}, d_i) \dots \subseteq C(b_i^{(K-1)}, d_i)$$

avec aussi :

$$CL(b_i^{(0)}, d_i) \subseteq \dots \subseteq CL(b_i^{(k-1)}, d_i) \subseteq CL(b_i^{(k)}, d_i) \dots \subseteq CL(b_i^{(K-1)}, d_i)$$

25 On note  $CL(b_i^{(k)}, d_i) \setminus CL(b_i^{(k-1)}, d_i)$  le complémentaire de  $CL(b_i^{(k-1)}, d_i)$  dans  $CL(b_i^{(k)}, d_i)$ . Son cardinal est égal à  $NL(b_i^{(k)}, d_i) - NL(b_i^{(k-1)}, d_i)$ .



L'obtention des mots de code  $m_i^{(k)}$  (avec  $0 \leq k < K$ ), résultats de la quantification du vecteur des coefficients de la bande  $i$  pour chacun des flux binaires  $k$ , se fait comme suit.

- 5      • Pour le flux binaire  $k=0$ , l'opération de quantification se fait de manière classique comme habituellement dans le codeur TDAC. Elle permet d'obtenir les paramètres,  $sign_q^{(0)}(i)$ ,  $L_i^{(0)}$  et  $r_i^{(0)}$  qui permettent de construire le mot de code  $m_i^{(0)}$ . On
- 10      détermine d'ailleurs dans cette même étape les vecteurs  $\tilde{Y}(i)$  et  $sign(i)$ . Ils sont stockés en mémoire, ainsi que la permutation correspondante  $perm(i)$ , pour être utilisés, le cas échéant, dans les étapes suivantes relatives aux autres flux binaires.
- 15      • Pour les flux binaires  $1 \leq k < K$ , on procède de manière incrémentale, de  $k=1$  jusqu'à  $k=K-1$ , en utilisant préférentiellement les étapes suivantes :

Si  $(b_i^{(k)} = b_i^{(k-1)})$  alors :

- 20      1. le mot de code, sur la bande  $i$ , de la trame du flux binaire  $k$  est le même que celui de la trame du flux binaire  $(k-1)$  :

$$\text{et } m_i^{(k)} = m_i^{(k-1)}$$

Sinon, i.e.  $(b_i^{(k)} > b_i^{(k-1)})$  :

- 25      2. On recherche parmi les  $(NL(b_i^{(k)}, d_i) - NL(b_i^{(k-1)}, d_i))$  leaders de  $CL(b_i^{(k)}, d_i) \setminus CL(b_i^{(k-1)}, d_i)$  le plus proche voisin de  $\tilde{Y}(i)$ ,

3. Avec le résultat de l'étape 2 et connaissant le plus proche voisin de  $\tilde{Y}(i)$  dans  $CL(b_i^{(k-1)}, d_i)$ , on teste si le plus proche voisin de  $\tilde{Y}(i)$  dans  $CL(b_i^{(k)}, d_i)$  est dans  $CL(b_i^{(k-1)}, d_i)$  (cas "Flag=0" ci-après) ou  $CL(b_i^{(k)}, d_i) \setminus CL(b_i^{(k-1)}, d_i)$  (cas "Flag=1" ci-après),

4. Si Flag=0 ((le leader le plus proche de  $\tilde{Y}(i)$  dans  $CL(b_i^{(k-1)}, d_i)$  est aussi son plus proche voisin dans  $CL(b_i^{(k)}, d_i)$ ) alors :

$$m_i^{(k)} = m_i^{(k-1)}$$

Si Flag=1 (le leader le plus proche de  $\tilde{Y}(i)$  dans  $CL(b_i^{(k)}, d_i) \setminus CL(b_i^{(k-1)}, d_i)$  trouvé à l'étape 2 est aussi son plus proche voisin dans  $CL(b_i^{(k)}, d_i)$ ), soit  $L_i^{(k)}$  son numéro (avec  $L_i^{(k)} \geq NL(b_i^{(k-1)}, d_i)$ ), alors on effectue les étapes ci-après :

a. Recherche du rang  $r_i^{(k)}$  de  $Y_q^{(k)}(i)$  (nouveau vecteur quantifié de  $Y(i)$  dans la classe du leader  $\tilde{Y}_q^{(k)}(i)$ ) par exemple par l'algorithme de Schalkwijk en utilisant  $perm(i)$ ,

b. Détermination de  $sign_q^{(k)}(i)$  en utilisant  $sign(i)$  et  $perm(i)$ ,

c. Détermination du mot de code  $m_i^{(k)}$  à partir de  $L_i^{(k)}$ ,  $r_i^{(k)}$  et  $sign_q^{(k)}(i)$ .

**\* Deuxième exemple de réalisation : application à un codeur par transformée de type MPEG-1 Layer I&II**

Le codeur MPEG-1 Layer I&II, présenté à la figure 6a, utilise un banc de filtres à 32 sous-bandes uniformes (bloc 61 de la figure 6a) pour réaliser la transformation temps/fréquence du signal audio d'entrée  $s_0$ . Les échantillons de sortie de chaque sous-bande sont regroupés, puis normalisés par un facteur d'échelle commun (déterminé par le bloc fonctionnel 67) avant d'être quantifiés (bloc 62). Le nombre de niveaux du quantificateur scalaire uniforme utilisé pour chaque sous-bande résulte d'une procédure d'allocation dynamique des bits (réalisée par le bloc 63). Cette procédure utilise un modèle psychoacoustique (bloc 64) pour déterminer la répartition des bits qui rend le bruit de quantification le moins perceptible possible. Les modèles d'audition proposés dans la norme se basent sur l'estimation du spectre obtenu par une transformée de Fourier rapide (FFT) du signal temporel d'entrée (réalisée par le bloc 65). En se référant à la figure 6b, la trame  $s_c$ , multiplexée par le bloc 66 de la figure 6a et qui est finalement transmise, contient, après un champ d'entête  $H_D$ , l'ensemble des échantillons des sous-bandes quantifiés  $E_{SB}$ , qui représentent l'information principale, et une information complémentaire utilisée pour l'opération de décodage constituée par les facteurs d'échelle  $F_E$  et l'allocation de bits  $A_1$ .

A partir de ce schéma de codage, la construction d'un codeur multi-débits, dans une application de l'invention,

peut être réalisée en mettant en commun les blocs fonctionnels suivants, en se référant à la figure 7 :

- Banc de filtres d'analyse 61
- Détermination des facteurs d'échelle 67
- 5    • Calcul 65 de la transformée de Fourier FFT
- Détermination des seuils de masquage suivant un modèle psychoacoustique 64.

Les deux blocs 64 et 65 fournissent déjà les rapports signal à masque (flèches *SMR* des figures 6a et 7),  
 10 utilisés pour la procédure d'allocation de bits (bloc 70 de la figure 7).

Dans cet exemple de réalisation tel que représenté sur la figure 7, il est possible de tirer profit de la procédure  
 15 utilisée pour l'allocation de bits pour la mettre aussi en commun, mais en ajoutant toutefois quelques modifications à l'allocation (bloc 70 d'allocation des bits de la figure 7). Seul le bloc fonctionnel de quantification 62\_0 à 62\_(K-1) est donc spécifique à chaque flux binaire  
 20 correspondant à un débit  $D_k$ ,  $0 \leq k \leq K-1$ . Il en va de même pour le bloc de multiplexage 66\_0 à 66\_(K-1).

#### \* Allocation des bits

Dans le codeur MPEG-1 Layer I&II, l'allocation se fait  
 25 préférentiellement par une succession d'étapes itératives comme suit.

*Etape 0* : Initialisation à zéro du nombres de bits  $b_i$  de chacune des sous-bandes  $i$ ,  $0 \leq i < M$ .

Etape 1 : Mise à jour de la fonction de distorsion  $NMR(i)$  (appelée "rapport bruit à masque", de l'anglais "Noise to Mask Ratio") sur chacune des sous-bandes :

$$NMR(i) = SMR(i) - SNR(b_i),$$

5 où  $SNR(b_i)$  est le rapport signal à bruit correspondant au quantificateur ayant un nombre de bits  $b_i$ , et  $SMR(i)$  le rapport signal à masque fourni par le modèle psychoacoustique.

10 Etape 2 : Incrémentation du nombre de bits  $b_{i_0}$  de la sous-bande  $i_0$  où cette distorsion est maximale:

$$b_{i_0} = b_{i_0} + \varepsilon, \quad i_0 = \arg \max_i [NMR(i)]$$

où  $\varepsilon$  est une valeur entière positive, dépendant de la bande, en général prise égale à 1.

15 Les étapes 1 et 2 sont répétées de manière itérative jusqu'à ce que le nombre total de bits disponibles, correspondant au débit de fonctionnement, soit distribué. Le résultat est alors un vecteur de distribution de bits

20  $(b_0, b_1, \dots, b_{M-1})$ .

Dans le schéma de codage multi-débits, ces étapes sont mises en commun avec quelques autres modifications, notamment :

25 • le bloc fonctionnel ayant pour sortie  $K$  vecteurs de distributions de bits  $(b_0^{(k)}, b_1^{(k)}, \dots, b_{M-1}^{(k)})$  (avec  $0 \leq k \leq K-1$ ), un vecteur  $(b_0^{(k)}, b_1^{(k)}, \dots, b_{M-1}^{(k)})$  est obtenu lorsque le nombre total de bits disponibles correspondant au débit

binaire  $D_k$  du flux binaire  $k$  est distribué, à l'itération des étapes 1 et 2.

- L'arrêt de l'itération des étapes 1 et 2 se fait lorsque le nombre total de bits disponibles correspondant au débit binaire le plus élevé  $D_{K-1}$  est totalement distribué (on rappelle que les flux binaires sont ordonnés suivant un ordre croissant de débits).

On notera que les vecteurs de distribution de bits sont obtenus successivement à partir de  $k=0$  jusqu'à  $k=K-1$ . Les  $K$  sorties de ce bloc d'allocation de bits alimentent alors les blocs de quantification pour chacun des flux binaires au débit donné.

**\* Troisième exemple de réalisation : application à un codeur de type CELP**

Le dernier exemple de réalisation concerne le codage de la parole multi-modes à décision a posteriori à partir du codeur 3GPP NB-AMR (pour "Narrow-Band Adaptive Multi-Rate") qui est un codeur de parole en bande téléphonique multi-débits adaptatif, selon une norme 3GPP. Ce codeur qui appartient à la famille bien connue des codeurs CELP dont le principe a été décrit brièvement plus haut, comporte huit modes (ou débits) allant de 12,2 kbit/s à 4,75 kbit/s, tous basés sur la technique ACELP (pour "Algebraic Code Excited Linear Prediction"). La figure 8 donne le schéma de codage en blocs fonctionnels de ce codeur. Cette structure a été exploitée afin de réaliser

un codeur multi-modes à décision a posteriori, basé sur 4 modes du codeur NB-AMR (7,4; 6,7; 5,9; 5,15).

5 Dans une première variante, seule la mutualisation des blocs fonctionnels identiques est exploitée (les résultats des quatre codages sont alors identiques à ceux des quatre codages en parallèle).

10 Dans une deuxième variante, la complexité est encore plus réduite. Les calculs de blocs fonctionnels non identiques pour certains modes sont accélérés en exploitant ceux d'un autre mode ou d'un module de traitement commun, comme on le verra ci-après. Les résultats des quatre codages ainsi mutualisés sont alors différents de ceux des quatre  
15 codages en parallèle.

Dans une autre variante encore, les blocs fonctionnels de ces quatre modes sont utilisés pour un codage multi-modes en treillis, comme on l'a vu ci-avant en référence à la  
20 figure 1d.

On rappelle brièvement ci-après les quatre modes (7,4; 6,7; 5,9; 5,15) du codeur 3GPP NB-AMR.

25 Le codeur 3GPP NB-AMR travaille sur un signal de parole limité en bande à 3,4 kHz et échantillonné à 8 kHz découpé en trames de 20 ms (160 échantillons). Chaque trame comporte 4 sous-trames de 5 ms (40 échantillons) regroupées 2 par 2 dans des "super sous-trames" de 10 ms  
30 (80 échantillons). Pour tous les modes, les mêmes types de paramètres sont extraits du signal mais avec des variantes

de modélisation et/ou de quantification de ces paramètres. Dans le codeur NB-AMR, cinq types de paramètres sont à analyser et à coder. Les paramètres LSP (pour "*Line Spectral Pairs*") sont traités une fois par trame pour tous les modes, sauf pour le mode 12,2 (donc une fois par super sous-trame). Les autres paramètres (notamment le retard LTP, le gain de l'excitation adaptative, l'excitation fixe, le gain de l'excitation fixe) sont traités une fois par sous-trame.

Les quatre modes considérés ici (7,4; 6,7; 5,9; 5,15) se distinguent essentiellement par les quantifications de leurs paramètres. L'allocation binaire de ces 4 modes est résumée dans le tableau 1 ci-après.

Mode (kbit/s)	7,4	6,7	5,9	5,15
LSP	26=(8+9+9)	26=(8+9+9)	26=(8+9+9)	23=(8+8+7)
Retards LTP	8 / 5 / 8 / 5	8 / 4 / 8 / 4	8 / 4 / 8 / 4	8 / 4 / 4 / 4
Excitation fixe	17/17/ 17/17	14/14/ 14/14	11/11/ 11/11	9 / 9 / 9 / 9
Gains des excitations fixe et adaptative	7 / 7 / 7 / 7	7 / 7 / 7 / 7	6 / 6 / 6 / 6	6 / 6 / 6 / 6
Total par trame	148	134	118	103

**Tableau 1:** Allocation binaire des 4 modes  
(7,4; 6,7; 5,9; 5,15) du codeur 3GPP NB-AMR

Ces 4 modes du codeur NB-AMR (7,4; 6,7; 5,9; 5,15) possèdent des modules identiques comme par exemple le pré-traitement, l'analyse des coefficients de prédiction linéaire, le calcul de signal pondéré. Le pré-traitement



du signal est un filtrage passe-haut de fréquence de  
 coupure 80 Hz pour supprimer les composantes continues  
 combiné à une division par deux des signaux d'entrée pour  
 éviter des débordements. L'analyse LPC comprend des sous-  
 5 modules de fenêtrage, de calcul des autocorrélations, de  
 mise en œuvre de l'algorithme de Levinson-Durbin, de  
 transformation  $A(z) \rightarrow LSP$ , de calcul des paramètres  $LSP_i$   
 non quantifiées pour chaque sous-trame ( $i=0, \dots, 3$ ) par  
 interpolation entre les LSP de la trame passée et ceux de  
 10 la trame courante, et de transformation inverse ( $LSP_i \rightarrow$   
 $A_i(z)$ ).

Le calcul du signal de parole pondéré réside en un  
 filtrage par le filtre de pondération perceptuelle:  
 15  $(W_i(z) = A_i(z/\gamma_1) / A_i(z/\gamma_2))$  où  $A_i(z)$  est le filtre non  
 quantifié de la sous-trame d'indice  $i$  avec  $\gamma_1 = 0,94$  et  
 $\gamma_2 = 0,6$ .

D'autres blocs fonctionnels ne sont identiques que pour  
 20 trois de ces modes (7,4; 6,7; 5,9). Par exemple, la  
 recherche du retard LTP en boucle ouverte effectuée sur le  
 signal pondéré une fois par super sous-trame pour ces  
 trois modes. Pour le mode à 5,15, elle n'est effectuée en  
 revanche qu'une fois par trame.

25 De même, si les quatre modes utilisent une quantification  
 vectorielle pondérée prédictive MA (pour "Moving Average")  
 d'ordre 1 à moyenne supprimée et par produit cartésien des  
 paramètres LSP dans le domaine fréquentiel normalisé, la  
 30 quantification des paramètres LSP du mode à 5,15 kbit/s se

fait sur 23 bits, celle des trois autres modes sur 26 bits. Après transformation dans le domaine fréquentiel normalisé, la quantification vectorielle par produit cartésien (dite "split VQ") des paramètres LSP scinde les

5 10 paramètres LSP en 3 sous-vecteurs, de dimension 3, 3 et 4. Le premier sous-vecteur composé des 3 premiers LSP est quantifié sur 8 bits par le même dictionnaire pour les quatre modes. Le deuxième sous-vecteur composé des 3 LSP

10 suivants est quantifié pour les 3 modes haut débit par un dictionnaire de taille 512 (9 bits) et pour le mode à 5,15 par la moitié de ce dictionnaire (un vecteur sur 2). Le troisième et dernier sous-vecteur composé des 4 derniers

15 LSP est quantifié pour les 3 modes haut débit par un dictionnaire de taille 512 (9 bits) et pour le mode de plus faible débit par un dictionnaire de taille 128 (7 bits). La transformation dans le domaine fréquentiel normalisé, le calcul des poids du critère d'erreur

20 quadratique et la prédiction MA (pour "Moving Average") du résidu LSP à quantifier sont identiques pour les 4 modes. Les trois modes haut débit utilisant les même

dictionnaires pour quantifier les LSP, ils peuvent partager, en plus du même module de quantification vectorielle, la transformation inverse (pour revenir du

25 domaine fréquentiel normalisé vers le domaine en cosinus), ainsi que le calcul des  $LSP_i^Q$  quantifiées pour chaque sous-trame ( $i=0,\dots,3$ ) par interpolation entre les LSP quantifiés de la trame passée et ceux de la trame courante, et enfin la transformation inverse  $LSP_i^Q \rightarrow A_i^Q(z)$ .

Les recherches en boucle fermée des excitations adaptative et fixe sont faites séquentiellement et nécessitent au préalable le calcul de la réponse impulsionnelle du filtre de synthèse pondéré, puis de signaux-cible. La réponse  
 5 impulsionnelle du filtre de synthèse pondéré  $(A_i(z/\gamma_1)/[A_i^0(z)A_i(z/\gamma_2)])$  est identique pour les 3 modes haut débit (7,4; 6,7; 5,9). Pour chaque sous-trame, le calcul du signal-cible pour l'excitation adaptative dépend du signal pondéré (indépendamment du mode), du filtre  
 10 quantifié  $A_i^0(z)$  (identique pour 3 des modes) et du passé de la sous-trame (différent pour chaque sous-trame autre que la première sous-trame). Pour chaque sous-trame, le signal-cible pour l'excitation fixe est obtenu en retirant au signal-cible précédent la contribution de l'excitation  
 15 adaptative filtrée de cette sous-trame (qui est différente d'un mode à l'autre sauf pour la première sous-trame des 3 premiers modes).

Trois dictionnaires adaptatifs sont utilisés. Le premier  
 20 dictionnaire, pour les sous-frames paires ( $i=0$  et 2) des modes (7,4; 6,7; 5,9) et pour la première sous-trame du mode à 5,15, comporte 256 retards absolus fractionnaires, de résolution  $1/3$  dans l'intervalle  $[19 + 1/3, 84 + 2/3]$  et de résolution entière dans l'intervalle  $[85, 143]$ . La  
 25 recherche dans ce dictionnaire de retards absolus est focalisée autour du retard trouvé en boucle ouverte (intervalle de  $\pm 5$  pour le mode à 5,15, de  $\pm 3$  pour les autres modes). Pour la première sous-trame des modes (7,4; 6,7; 5,9), le signal-cible et le retard en boucle ouverte  
 30 étant identique, le résultat de cette recherche en boucle fermée l'est aussi. Les deux autres dictionnaires sont de

type différentiel et permettent de coder la différence entre le retard courant et le retard entier  $T_{i-1}$  le plus proche du retard fractionnaire de la sous-trame précédente. Le premier dictionnaire différentiel sur 5 bits, utilisé pour les sous-frames impaires du mode à 7,4, est de résolution  $1/3$  autour du retard entier  $T_{i-1}$  dans l'intervalle  $[T_{i-1}-5 + 2/3, T_{i-1}+4 + 2/3]$ . Le deuxième dictionnaire différentiel sur 4 bits, inclus dans le premier, est utilisé pour les sous-frames impaires des modes à 6,7 et 5,9 ainsi que pour les trois dernières sous-frames du mode à 5,15. Ce deuxième dictionnaire est de résolution entière autour du retard entier  $T_{i-1}$  dans l'intervalle  $[T_{i-1}-5, T_{i-1}+4]$  plus une résolution de  $1/3$  dans l'intervalle  $[T_{i-1}-1 + 2/3, T_{i-1} + 2/3]$ .

Les dictionnaires fixes appartiennent à la famille bien connue des dictionnaires ACELP. La structure d'un répertoire ACELP est basée sur le concept ISPP (pour "*Interleaved Single-Pulse Permutation*") qui consiste à diviser l'ensemble des  $L$  positions en  $K$  pistes entrelacées, chacune des  $N$  impulsions étant localisée dans certaines pistes prédéfinies. Les 4 modes (7,4; 6,7; 5,9; 5,15) utilisent la même découpe des 40 échantillons d'une sous-trame en 5 pistes de longueur 8 entrelacées, comme le montre le tableau 2a. Le tableau 2b montre, quant à lui, pour les 3 modes (7,4; 6,7; 5,9) le débit du dictionnaire, le nombre d'impulsions et leur répartition dans les pistes. La répartition des 2 impulsions du dictionnaire ACELP à 9 bits du mode à 5,15 est encore plus contrainte.

Piste	Positions
P <sub>0</sub>	0, 5, 10, 15, 20, 25, 30, 35
P <sub>1</sub>	1, 6, 11, 16, 21, 26, 31, 36
P <sub>2</sub>	2, 7, 12, 17, 22, 27, 32, 37
P <sub>3</sub>	3, 8, 13, 18, 23, 28, 33, 38
P <sub>4</sub>	4, 9, 14, 19, 24, 29, 34, 39

**Tableau 2a:** Découpe en pistes entrelacées des 40 positions d'une sous-trame du codeur 3GPP NB-AMR

5

Mode (kbit/s)	7,4	6,7	5,9
Débit du dictionnaire ACELP (positions+amplitudes)	17 (13+4)	14 (11+3)	11 (9+2)
Nombre d'impulsions	4	3	2
Pistes potentielles pour i <sub>0</sub>	P <sub>0</sub>	P <sub>0</sub>	P <sub>1</sub> , P <sub>3</sub>
Pistes potentielles pour i <sub>1</sub>	P <sub>1</sub>	P <sub>1</sub> , P <sub>3</sub>	P <sub>0</sub> , P <sub>1</sub> , P <sub>2</sub> , P <sub>4</sub>
Pistes potentielles pour i <sub>2</sub>	P <sub>2</sub>	P <sub>2</sub> , P <sub>4</sub>	-
Pistes potentielles pour i <sub>3</sub>	P <sub>3</sub> , P <sub>4</sub>	-	-

**Tableau 2b:** Répartition des impulsions dans les pistes pour les modes 7,4; 6,7; 5,9 du codeur 3GPP NB-AMR

Les gains des excitations adaptative et fixe sont quantifiés sur 7 ou 6 bits (avec une prédiction MA appliquée au gain de l'excitation fixe) par une  
 5 quantification vectorielle conjointe minimisant le critère CELP.

*\* Codage multi-modes à décision a posteriori n'exploitant que la mutualisation des blocs fonctionnels identiques*

10 A partir de ce schéma de codage, la construction d'un codeur multi-modes à décision a posteriori peut être réalisée en mettant en commun les blocs fonctionnels suivants.

En se référant à la figure 8, pour les 4 modes, on effectue en commun :

- le pré-traitement (bloc 81),
- l'analyse des coefficients de prédiction linéaire (fenêtrage et calcul des autocorrélations 82, mise en œuvre de l'algorithme de Levinson-Durbin 83, transformation  $A(z) \rightarrow \text{LSP}$  84, interpolation des LSP et transformation inverse 862),
- le calcul du signal d'entrée pondéré 87,
- la transformation des paramètres LSP dans le domaine fréquentiel normalisé, le calcul des poids du critère d'erreur quadratique pour la quantification vectorielle des LSP, la prédiction MA du résidu LSP, la quantification vectorielle des 3 premiers LSP (dans le bloc 85).

Pour tous ces blocs, leur complexité cumulée est ainsi  
 30 divisée par 4.

Pour les 3 modes de plus haut débit (7,4; 6,7; 5,9), on effectue :

- la quantification vectorielle des 7 derniers LSP (une fois par trame) (dans le bloc 85 de la figure 8),
- 5     • la recherche du retard LTP en boucle ouverte (2 fois par trame) (bloc 88),
- l'interpolation des LSP quantifiés (861) et la transformation inverse vers les filtres  $A^0_i$  (pour chaque sous-trame),
- 10    • le calcul de la réponse impulsionnelle 89 du filtre de synthèse pondéré (pour chaque sous-trame).

Pour ces blocs, les calculs ne sont plus effectués 4 fois mais 2 fois, une fois pour les 3 modes à plus haut débit et une fois pour le mode à faible débit. Leur complexité  
15 est donc divisée par 2.

On peut aussi, pour ces 3 modes de plus haut débit, mutualiser pour la première sous-trame le calcul des signaux-cible pour l'excitation fixe (bloc 91 sur la  
20 figure 8) et adaptative (bloc 90), ainsi que la recherche LTP en boucle fermée (bloc 881). Il faut noter que la mutualisation de ces opérations pour la première sous-trame ne produit des résultats identiques que dans le contexte du codage multiple de type multi-modes à décision  
25 à posteriori. Dans le contexte général de codage multiple, le passé de la première sous-trame est, comme pour les 3 autres sous-frames, différent selon les débits, ces opérations conduisent généralement alors à des résultats différents.

\* *Codage multi-modes à décision a posteriori avancée*

Des blocs fonctionnels non identiques peuvent être accélérés en exploitant ceux d'un autre mode ou d'un module de traitement commun. Selon les contraintes de l'application (en termes de qualité et/ou de complexité), on peut utiliser différentes variantes. Quelques exemples sont décrits ci-après. Il est aussi possible de s'appuyer sur des techniques de transcodage intelligent entre codeurs CELP.

\* La quantification vectorielle du deuxième sous-vecteur de LSP

On peut, comme dans le cas du mode de réalisation pour le codeur TDAC, exploiter l'imbrication de certains dictionnaires pour accélérer les calculs. Ainsi, le dictionnaire du deuxième sous-vecteur de LSP du mode à 5,15 étant inclus dans celui des 3 autres modes, la quantification de ce sous-vecteur Y par les 4 modes peut être ainsi avantageusement combinée:

- Etape 1: Chercher son plus proche voisin  $Y_1$  dans le plus petit dictionnaire (correspondant à la moitié du grand dictionnaire)
  - o  $Y_1$  quantifie Y pour le mode à 5,15
- Etape 2: Chercher le plus proche voisin  $Y_h$  dans le complémentaire dans le grand dictionnaire (soit l'autre moitié du dictionnaire)
- Etape 3: Tester si le plus proche voisin de Y dans le dictionnaire à 9 bits est  $Y_1$  (cas "Flag=0") ou  $Y_h$  (cas "Flag=1")



- o cas "Flag=0" :  $Y_1$  quantifie aussi  $Y$  pour les modes à 7,4; 6,7 et 5,9
- o sinon (cas "Flag=1"),  $Y_h$  quantifie  $Y$  pour les modes à 7,4; 6,7 et 5,9

5 Cette mise en œuvre donne un résultat identique à celui du codage multi-mode non optimisé. Si l'on désire réduire davantage la complexité de la quantification, on peut s'arrêter à l'étape 1 et prendre  $Y_1$  comme vecteur quantifié pour les modes haut débit si ce vecteur est jugé  
 10 suffisamment proche de  $Y$ . Cette simplification peut donc donner un résultat différent d'une recherche exhaustive.

\* Accélération de la recherche LTP en boucle ouverte  
 La recherche du retard LTP en boucle ouverte du mode à 5,15 peut exploiter les résultats de celle des autres  
 15 modes. Si les deux retards en boucle ouverte trouvés sur les 2 super sous-trames sont suffisamment proches pour permettre un codage différentiel, la recherche en boucle ouverte du mode à 5,15 n'est pas effectuée. On utilise plutôt les résultats des modes supérieurs. Sinon, on  
 20 peut :

- effectuer la recherche classique,
- ou focaliser la recherche en boucle ouverte sur toute la trame autour des deux retards en boucle ouverte trouvés par les modes supérieurs.

25 A l'inverse, on peut aussi effectuer d'abord la recherche du retard en boucle ouverte sur le mode à 5,15 et focaliser les deux recherches du retard en boucle ouverte des modes supérieurs autour de la valeur déterminée par le  
 30 mode à 5,15.

Dans une troisième variante plus avancée, illustrée à la figure 1d, on se propose de réaliser un codeur multi-modes en treillis permettant plusieurs combinaisons de blocs fonctionnels, chaque bloc fonctionnel possédant au moins deux modes de fonctionnement (ou débits). On construit ce nouveau codeur à partir des quatre débits du codeur NB-AMR cités ci-avant (5,15; 5,90; 6,70; 7,40). Dans ce codeur, on distingue quatre blocs fonctionnels: le bloc LPC, le bloc LTP, le bloc excitation fixe et le bloc de gains. En se référant au tableau 1 présenté ci-avant, le tableau 3a ci-après récapitule pour chacun de ces blocs fonctionnels, son nombre de débits et ses débits.

Bloc fonctionnel	Nombre de débits	Débits des blocs fonctionnels
LPC (LSP)	2	26 et 23
Retard LTP	3	26, 24 et 20
Excitation fixe	4	68, 56, 44 et 36
Gains	2	28 et 24

**Tableau 3a:** Nombre de débits et débits des blocs fonctionnels pour les quatre modes (5,15; 5,90; 6,70; 7,40) du codeur NB-AMR.

On a donc  $P=4$  blocs fonctionnels et  $2 \times 3 \times 4 \times 2 = 48$  combinaisons possibles. Dans l'exemple particulier de réalisation, on choisit de ne pas considérer le haut-débit du bloc fonctionnel 2 (LTP débit 26 bits/trame). Un autre choix est possible, bien entendu.

Le codeur multi-débits ainsi obtenu possède une grande granularité en débits, avec 32 modes possibles donnés dans

le tableau 3b. Toutefois, on indique que le codeur ainsi obtenu n'est pas interopérable avec le codeur NB-AMR précité. Dans le tableau 3b, les modes correspondants aux trois débits du NB-AMR (5,15; 5,90; 6,70) sont présentés en gras, l'exclusion du débit le plus élevé du bloc fonctionnel LTP éliminant le débit de 7,40.

Paramètres	LSP	Retard LTP	Excitation fixe	Gains des excitations fixe et adaptative	Total
Débit par trame	<b>23</b>	<b>20</b>	<b>36</b>	<b>24</b>	<b>103</b>
	23	20	36	28	107
	23	20	44	24	111
	23	20	44	28	115
	23	20	56	24	123
	23	20	56	28	127
	23	20	68	24	135
	23	20	68	28	139
	23	24	36	24	107
	23	24	36	28	111
	23	24	44	24	115
	23	24	44	28	119
	23	24	56	24	127
	23	24	56	28	131
	23	24	68	24	139
	23	24	68	28	143
	26	20	36	24	106
	26	20	36	28	110
	26	20	44	24	114

	26	20	44	28	118
	26	20	56	24	126
	26	20	56	28	130
	26	20	68	24	138
	26	20	68	28	142
	26	24	36	24	110
	26	24	36	28	114
	<b>26</b>	<b>24</b>	<b>44</b>	<b>24</b>	<b>118</b>
	26	24	44	28	122
	26	24	56	24	130
	<b>26</b>	<b>24</b>	<b>56</b>	<b>28</b>	<b>134</b>
	26	24	68	24	142
	26	24	68	28	146

**Tableau 3b:** Débit par bloc fonctionnel et global du codeur multi-modes en treillis

Ce codeur possédant 32 débits possibles, 5 bits sont nécessaires pour identifier le mode utilisé. Comme dans la variante précédente, la mutualisation de blocs fonctionnels est exploitée. On applique des stratégies de codage différentes pour les différents blocs fonctionnels. Par exemple, pour le bloc fonctionnel 1 comprenant la quantification des LSP, on privilégie le bas débit comme mentionné ci-avant de la manière suivante:

- Le premier sous-vecteur composé des 3 premiers LSP est quantifié sur 8 bits par le même dictionnaire pour les deux débits associés à ce bloc fonctionnel,
- Le deuxième sous-vecteur composé des 3 LSP suivants est quantifié sur 8 bits par le dictionnaire du plus petit débit. Ce dictionnaire correspondant à la moitié du dictionnaire de plus haut débit, on n'effectue la

recherche dans l'autre moitié du dictionnaire que si la distance entre les 3 LSP et l'élément choisi dans le dictionnaire dépasse un certain seuil.

- Le troisième et dernier sous-vecteur composé des 4 derniers LSP est quantifié par un dictionnaire de taille 512 (9 bits) et par un dictionnaire de taille 128 (7 bits).

Par contre, comme mentionné ci-avant dans la deuxième variante (correspondant au codage multi-modes à décision a posteriori avancée), on choisit de privilégier le haut-débit pour le bloc fonctionnel 2 (retard LTP). Dans le codeur NB-AMR, la recherche du retard LTP en boucle ouverte est effectuée deux fois par trame pour le retard LTP de 24 bits et elle est effectuée une seule fois par trame pour celui de 20 bits. Pour ce bloc fonctionnel, on souhaite favoriser le haut débit. Donc, le calcul du retard LTP en boucle ouverte est réalisé de la manière suivante:

- On calcule deux retards en boucle ouverte sur les 2 super sous-trames. S'ils sont suffisamment proches pour permettre un codage différentiel, la recherche en boucle ouverte sur la trame entière n'est pas effectuée. On utilise plutôt les résultats des deux super sous-trames.
- Sinon, on effectue une recherche en boucle ouverte sur toute la trame en la focalisant autour des deux retards en boucle ouverte trouvés précédemment. Une variante réduisant la complexité retient uniquement le retard en boucle ouverte de la première.

Après certains blocs fonctionnels, il est possible de réaliser une sélection partielle permettant de réduire le nombre de combinaisons à explorer. Par exemple, après le bloc fonctionnel 1 (LPC), on peut éliminer les  
5 combinaisons avec 26 bits pour ce bloc si la performance du débit de 23 bits est suffisamment proche ou inversement éliminer le mode à 23 bits si sa performance est trop dégradée par rapport au mode à 26 bits.

10 Ainsi, la présente invention permet de fournir une solution efficace au problème de la complexité des codages multiples, par la mutualisation et l'accélération des calculs mis en œuvre par les différents codeurs. Les structures de codage peuvent donc être représentées à  
15 l'aide de blocs fonctionnels décrivant les différentes opérations effectuées au cours d'un traitement. Les blocs fonctionnels des différents codages mis en œuvre dans un codage multiple possèdent des relations fortes qui sont exploitées au sens de la présente invention. Ces relations  
20 sont particulièrement fortes lorsque les différents codages correspondent à différents modes d'une même structure.

On indique enfin que la présente invention est flexible du  
25 point de vue de la complexité. Il est possible en effet de décider a priori la complexité maximum du codage multiple et d'adapter le nombre de codeurs explorés en fonction de cette complexité.

### Revendications

1. Procédé de codage multiple en compression, dans lequel un signal d'entrée est destiné à alimenter en parallèle  
5 une pluralité de codeurs comportant chacun une succession de blocs fonctionnels, en vue d'un codage en compression dudit signal par chaque codeur, caractérisé en ce qu'il comporte les étapes préparatoires ci-après :
- 10 a) identifier les blocs fonctionnels formant chaque codeur, ainsi qu'une ou plusieurs fonctions réalisées par chaque bloc,  
b) repérer, parmi lesdites fonctions, des fonctions qui sont communes d'un codeur à l'autre, et  
15 c) exécuter lesdites fonctions communes, une fois pour toutes, pour une partie au moins de tous les codeurs, au sein d'au moins un même module de calcul.
2. Procédé selon la revendication 1, caractérisé en ce que  
20 ledit module de calcul est constitué par un ou plusieurs blocs de l'un des codeurs.
3. Procédé selon la revendication 2, caractérisé en ce que, pour chaque fonction exécutée à l'étape c), on  
25 utilise au moins un bloc fonctionnel d'un codeur choisi parmi ladite pluralité de codeurs, et en ce que le bloc dudit codeur choisi est agencé pour délivrer des résultats partiels aux autres codeurs, pour un codage efficace, auprès desdits autres codeurs,  
30 vérifiant un critère optimal entre la complexité et la qualité du codage.

4. Procédé selon la revendication 3, dans lequel les codeurs sont susceptibles d'opérer à des débits respectifs différents, caractérisé en ce que le codeur choisi est le codeur de débit le plus faible, et en ce que les résultats obtenus, suite à l'exécution de la fonction à l'étape c) avec des paramètres propres au codeur choisi, sont adaptés aux débits d'une partie au moins des autres codeurs par une recherche focalisée de paramètres pour une partie au moins de tous les autres modes, jusqu'au codeur de débit le plus élevé.

5. Procédé selon la revendication 3, dans lequel les codeurs sont susceptibles d'opérer à des débits respectifs différents, caractérisé en ce que le codeur choisi est le codeur de débit le plus élevé, et en ce que les résultats obtenus, suite à l'exécution de la fonction à l'étape c) avec des paramètres propres au codeur choisi, sont adaptés aux débits d'une partie au moins des autres codeurs par une recherche focalisée de paramètres pour une partie au moins de tous les autres modes, jusqu'au codeur de débit le plus faible.

6. Procédé selon la revendication 4, prise en combinaison avec la revendication 5, caractérisé en ce que, pour un débit donné, on utilise le bloc fonctionnel d'un codeur opérant audit débit donné, en tant que module de calcul, et on adapte progressivement au moins une partie des paramètres propres à ce codeur :  
- jusqu'au codeur de débit le plus élevé, par recherche focalisée, et



- jusqu'au codeur de débit le plus faible, par recherche focalisée.

5 7. Procédé selon la revendication 1, dans lequel les blocs fonctionnels des différents codeurs sont agencés en treillis, avec plusieurs chemins possibles dans le treillis, caractérisé en ce que chaque chemin du treillis est défini par une combinaison de modes de fonctionnement des blocs fonctionnels, chaque bloc fonctionnel alimentant  
10 plusieurs variantes possibles du bloc fonctionnel suivant.

8. Procédé selon la revendication 7, caractérisée en ce que l'on prévoit un module de sélection partielle, après chaque étape de codage menée par un ou plusieurs blocs  
15 fonctionnels, capable de sélectionner les résultats fournis par un ou plusieurs de ces blocs fonctionnels, pour des étapes suivantes de codage.

9. Procédé selon la revendication 7, dans lequel les blocs  
20 fonctionnels sont susceptibles d'opérer à des débits respectifs différents et en utilisant des paramètres respectifs propres auxdits débits, caractérisé en ce que, pour un bloc fonctionnel donné, le chemin du treillis choisi est celui traversant le bloc  
25 fonctionnel de débit le plus faible, et en ce que les résultats obtenus dudit bloc fonctionnel de débit le plus faible sont adaptés aux débits d'une partie au moins des autres blocs fonctionnels par une recherche focalisée de paramètres pour une partie au moins  
30 de tous les autres blocs fonctionnels, jusqu'au bloc fonctionnel de débit le plus élevé.

10. Procédé selon la revendication 7, dans lequel les blocs fonctionnels sont susceptibles d'opérer à des débits respectifs différents et en utilisant des paramètres respectifs propres auxdits débits,  
5 caractérisé en ce que, pour un bloc fonctionnel donné, le chemin du treillis choisi est celui traversant le bloc fonctionnel de débit le plus élevé,  
et en ce que les résultats obtenus dudit bloc fonctionnel de débit le plus élevé sont adaptés aux débits d'une  
10 partie au moins des autres blocs fonctionnels par une recherche focalisée de paramètres pour une partie au moins de tous les autres blocs fonctionnels, jusqu'au bloc fonctionnel de débit le plus faible.

11. Procédé selon la revendication 9, prise en combinaison avec la revendication 10, caractérisé en ce que, pour un débit donné associé aux paramètres d'un bloc fonctionnel d'un codeur, on utilise le bloc fonctionnel opérant audit  
15 débit donné, en tant que module de calcul, et on adapte progressivement au moins une partie des paramètres propres à ce bloc fonctionnel:

- jusqu'au bloc fonctionnel capable d'opérer au débit le plus faible, par recherche focalisée, et
- 25 - jusqu'au bloc fonctionnel capable d'opérer au débit le plus élevé, par recherche focalisée.

12. Procédé selon la revendication 1, caractérisé en ce que ledit module de calcul est un module indépendant  
30 desdits codeurs, et agencé pour redistribuer des résultats obtenus à l'étape c) à tous les codeurs.

13. Procédé selon la revendication 12, prise en combinaison avec la revendication 2, caractérisé en ce que le module indépendant et le ou les blocs de l'un au moins des codeurs sont agencés pour échanger mutuellement des résultats obtenus à l'étape c), et en ce que le module de calcul est agencé pour effectuer un transcodage d'adaptation entre blocs fonctionnels de codeurs différents.

10

14. Procédé selon l'une des revendications 12 et 13, caractérisé en ce que le module indépendant comporte un bloc de codage au moins partiel et un bloc de transcodage d'adaptation.

15

15. Procédé selon l'une des revendications précédentes, dans lequel les codeurs en parallèle sont agencés pour opérer en codage multi-modes, caractérisé en ce que l'on prévoit un module de sélection a posteriori, capable de sélectionner un codeur parmi les codeurs.

20

16. Procédé selon la revendication 15, caractérisé en ce que l'on prévoit un module de sélection partielle, après chaque étape de codage menée par un ou plusieurs blocs fonctionnels, indépendant des codeurs et capable de sélectionner un ou plusieurs codeurs.

25

17. Procédé selon l'une des revendications précédentes, dans lequel les codeurs sont de type par transformée, caractérisé en ce que le module de calcul comporte un bloc d'allocation de bits, partagé entre tous les codeurs,

30

chaque allocation de bits effectuée pour un codeur étant suivie d'une adaptation à ce codeur notamment en fonction de son débit.

- 5 18. Procédé selon la revendication 17, caractérisé en ce que le procédé comporte en outre une étape de quantification, dont les résultats sont fournis à tous les codeurs.
- 10 19. Procédé selon la revendication 18, caractérisé en ce qu'il comporte en outre des étapes communes à tous les codeurs parmi :
- une transformée temps-fréquence (MDCT),
  - une détection de voisement dans le signal d'entrée,
  - 15 - une détection de tonalité,
  - la détermination d'une courbe de masquage,
  - et un codage d'enveloppe spectrale.
- 20 20. Procédé selon la revendication 17, dans lequel les codeurs effectuent un codage en sous-bande (MPEG-1), caractérisé en ce que le procédé comporte en outre des étapes communes à tous les codeurs parmi :
- l'application d'un banc de filtres d'analyse,
  - une détermination de facteurs d'échelle,
  - 25 - un calcul de transformée spectrale (FFT),
  - et la détermination de seuils de masquage suivant un modèle psychoacoustique.

21. Procédé selon l'une des revendications 1 à 16, dans lequel les codeurs sont du type à analyse par synthèse (CELP), caractérisé en ce que le procédé comporte des étapes communes à tous les codeurs parmi au moins :

- 5    - un pré-traitement,
- l'analyse de coefficients de prédiction linéaire,
- un calcul de signal d'entrée pondéré,
- et une quantification pour au moins une partie des paramètres.

10

22. Procédé selon la revendication 21, prise en combinaison avec la revendication 16, caractérisé en ce que le module de sélection partielle est mis en œuvre après une étape partagée de quantification vectorielle pour des paramètres à court terme (LPC).

15

23. Procédé selon la revendication 21, prise en combinaison avec la revendication 16, caractérisé en ce que le module de sélection partielle est mis en œuvre après une étape partagée de recherche de paramètre à long terme (LTP) en boucle ouverte.

20

24. Produit programme d'ordinateur destiné à être stocké dans une mémoire d'une unité de traitement, notamment d'un ordinateur ou d'un terminal mobile, ou sur un support mémoire amovible et destiné à coopérer avec un lecteur de l'unité de traitement,

25

caractérisé en ce qu'il comporte des instructions pour la mise en œuvre du procédé de transcodage selon l'une des revendications précédentes.

- 5 25. Dispositif d'aide à un codage multiple en compression, codage dans lequel un signal d'entrée est destiné à alimenter en parallèle une pluralité de codeurs comportant chacun une succession de blocs fonctionnels, en vue d'un codage en compression dudit signal par chaque codeur,
- 10 caractérisé en ce qu'il comporte une mémoire propre à stocker des instructions d'un produit programme d'ordinateur selon la revendication 24.

26. Dispositif selon la revendication 25, caractérisé en
- 15 ce qu'il comporte en outre un module de calcul indépendant (MI) pour la mise en œuvre du procédé selon l'une des revendications 12 à 16 et 22, 23.

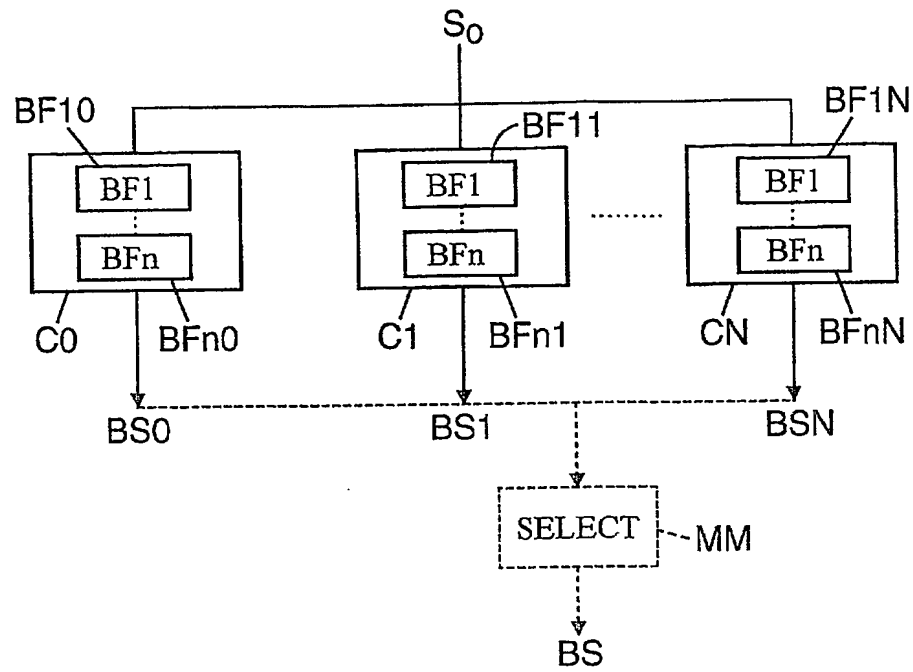


FIG. 1a

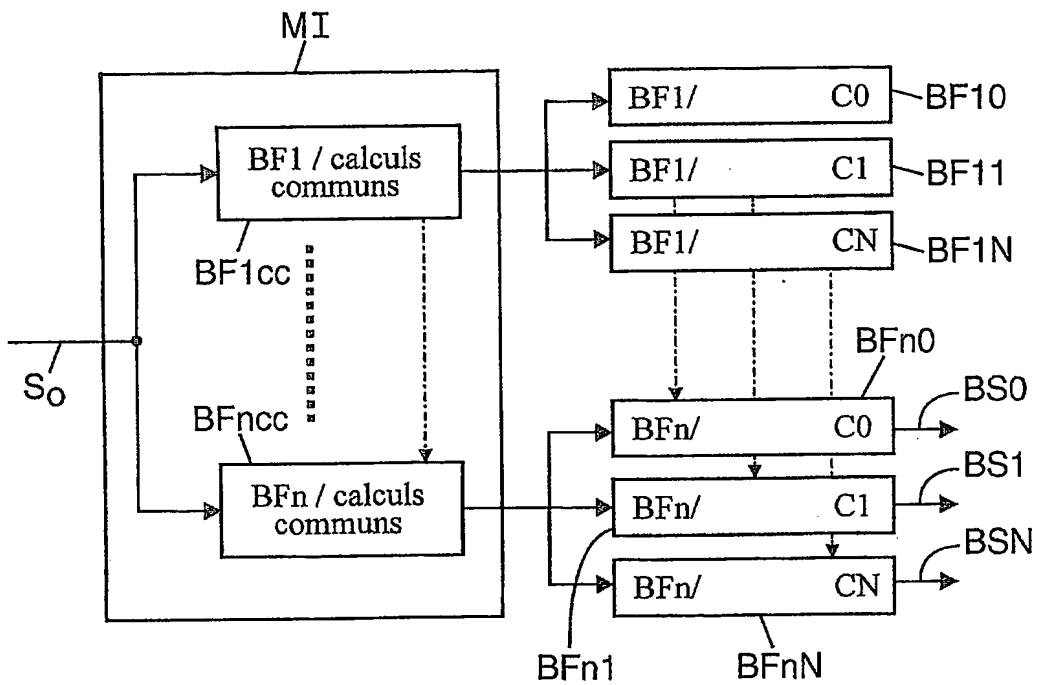


FIG. 1b

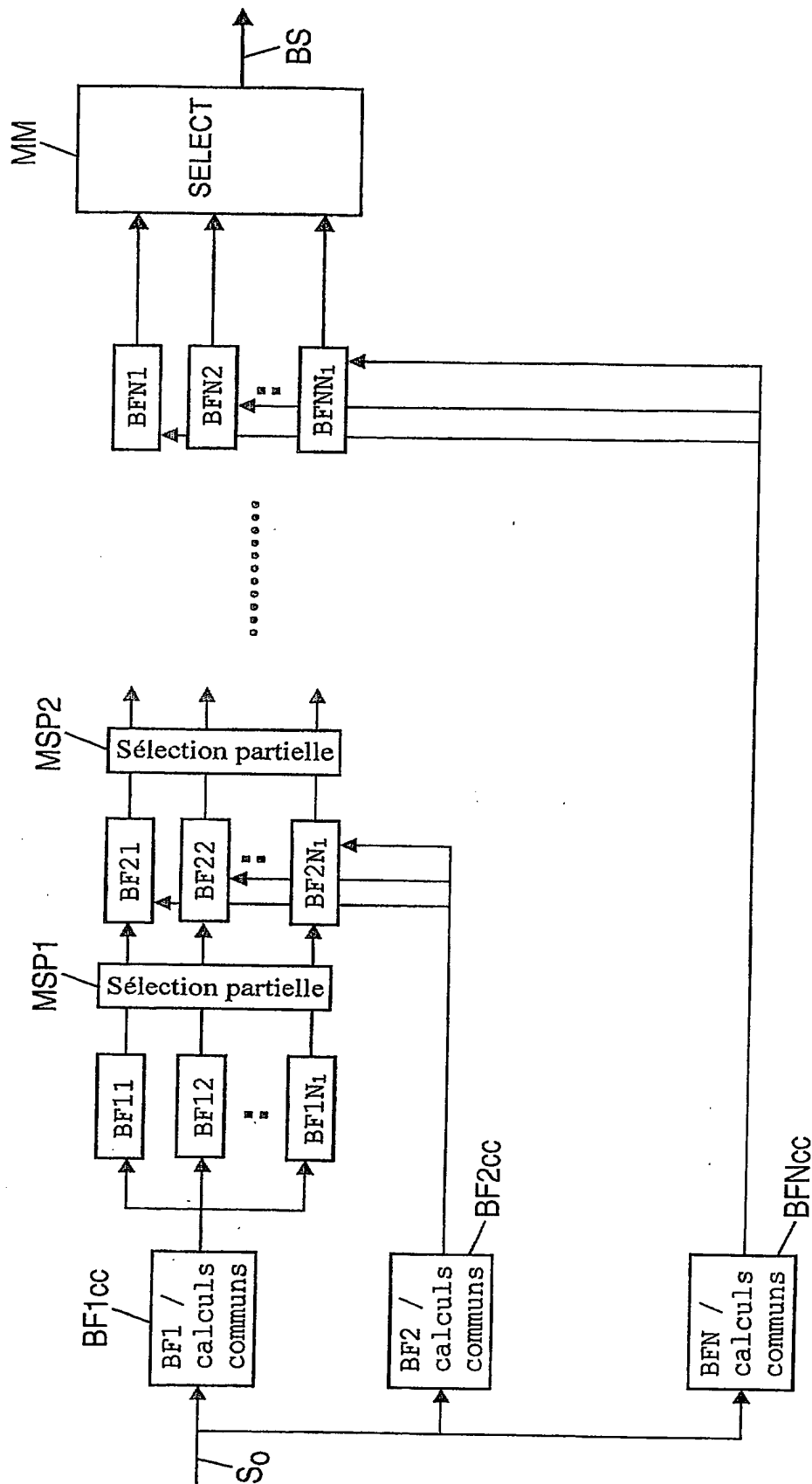


FIG. 1c





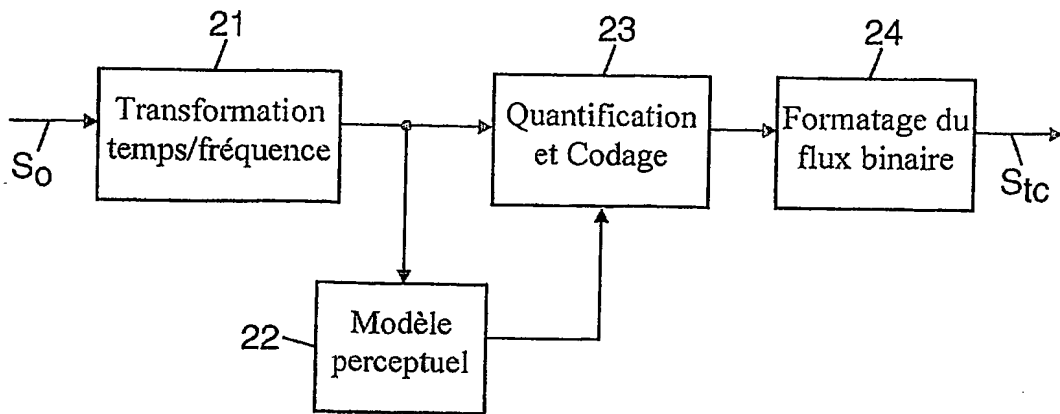


FIG. 2

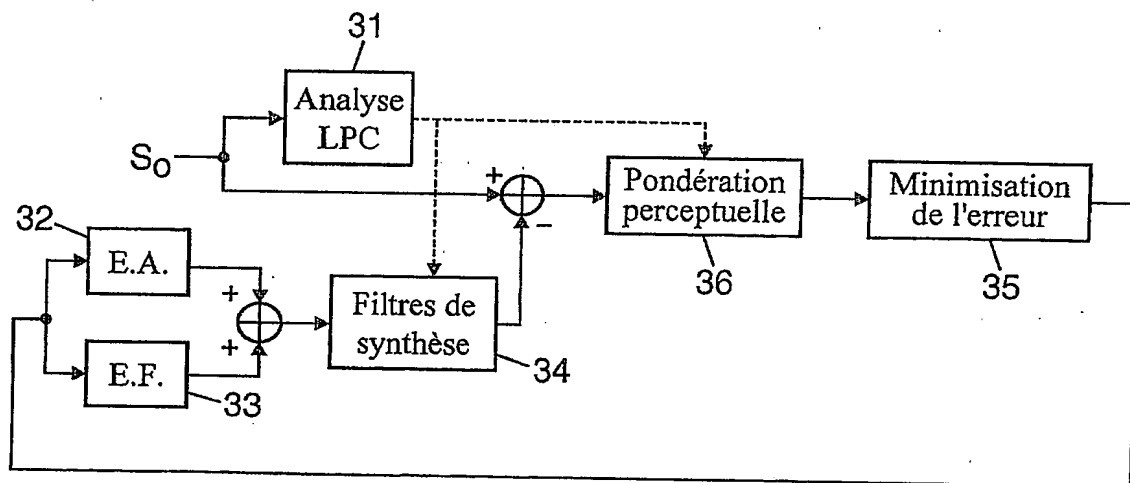


FIG. 3

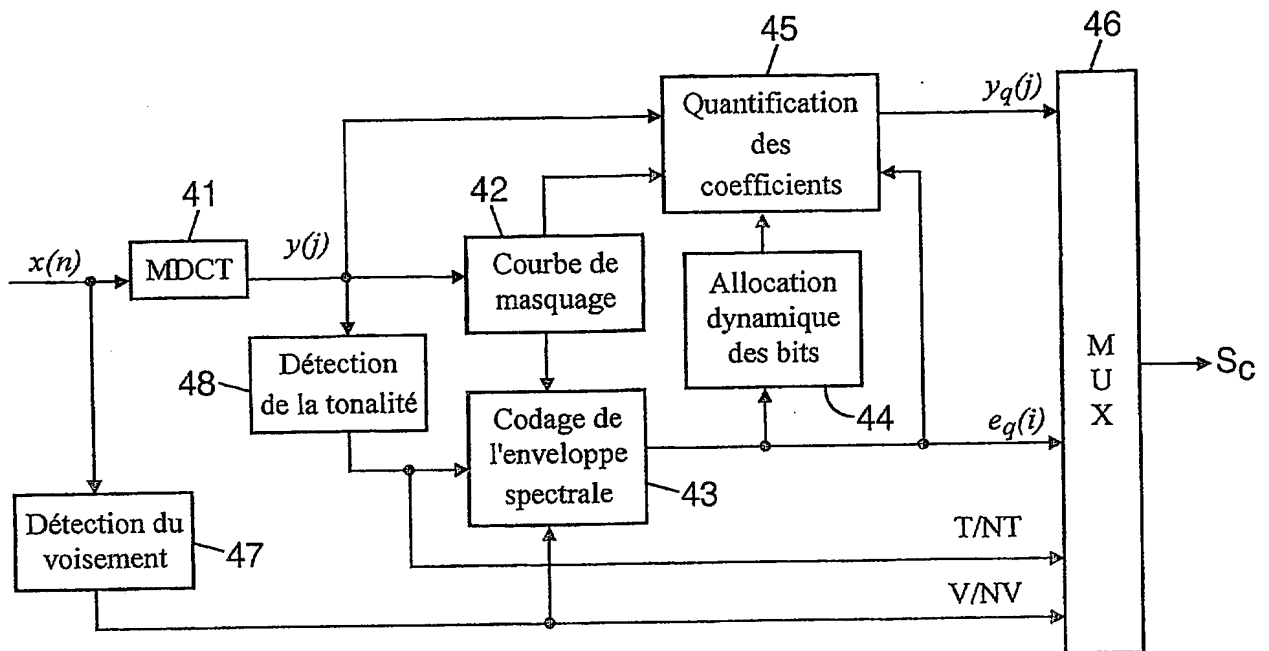


FIG. 4a

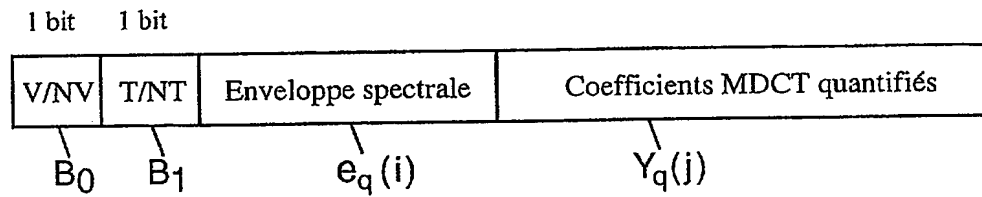


FIG. 4b

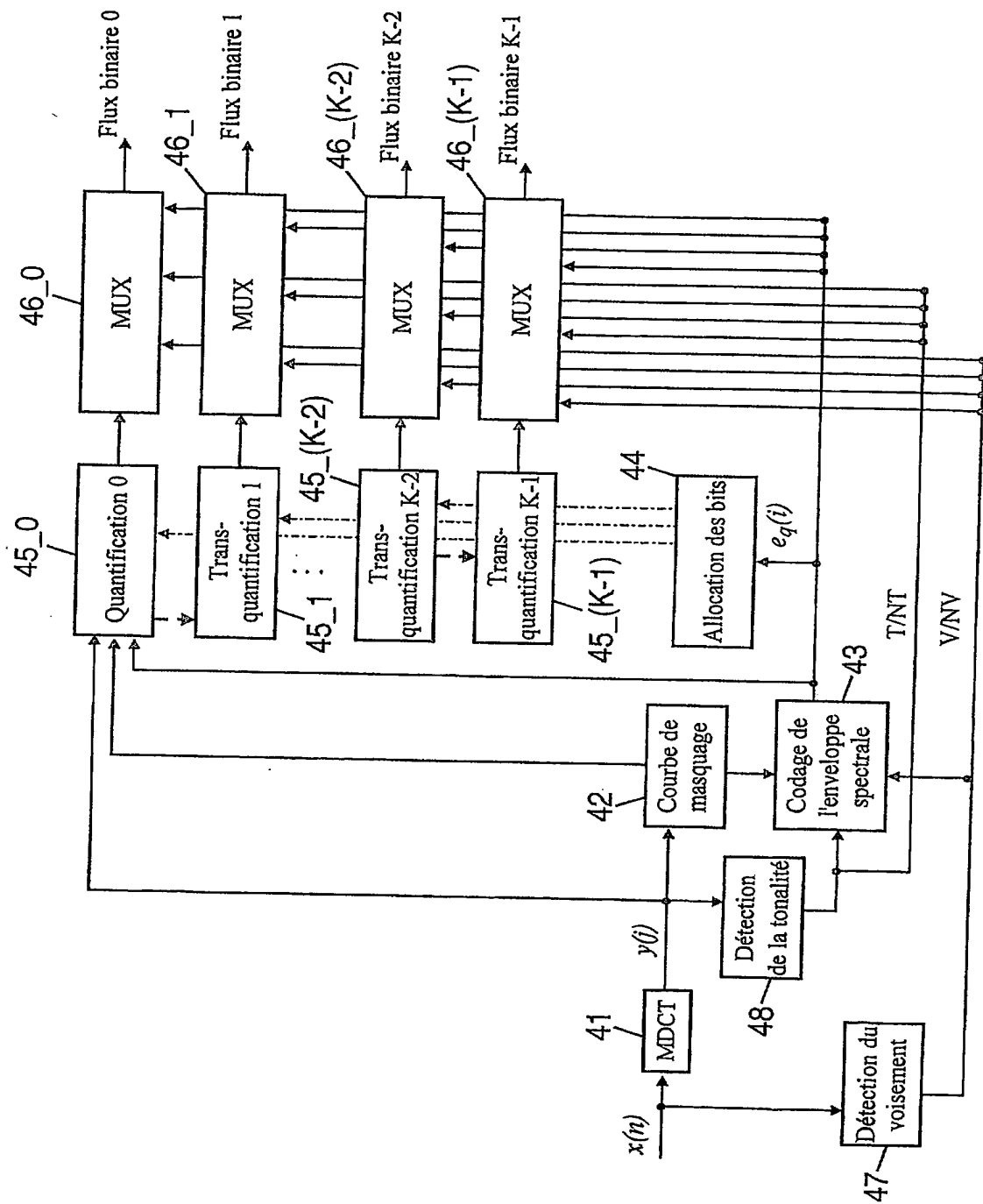


FIG. 5

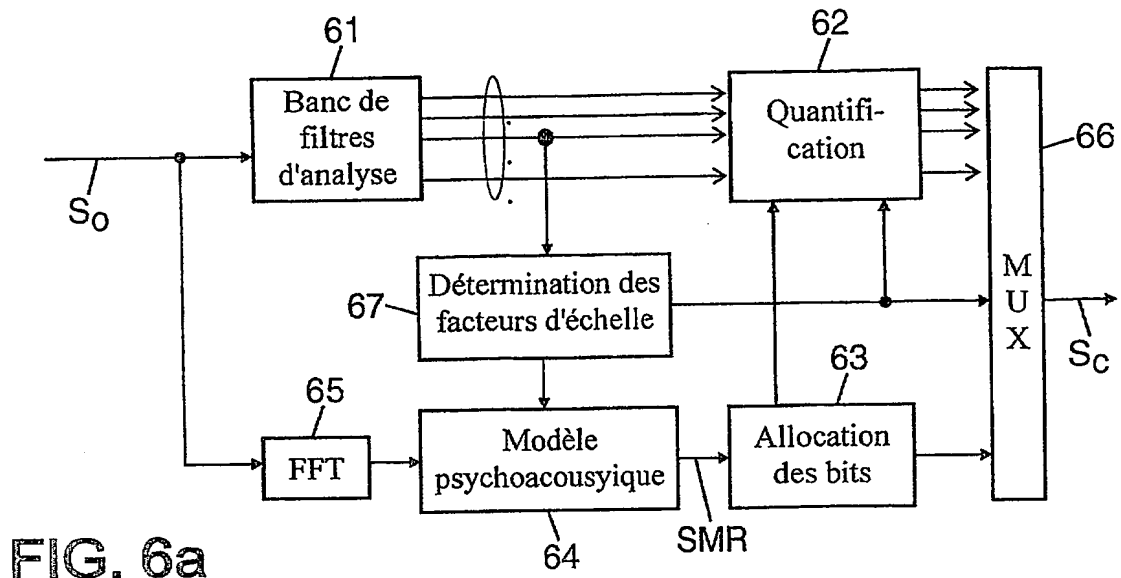


FIG. 6a

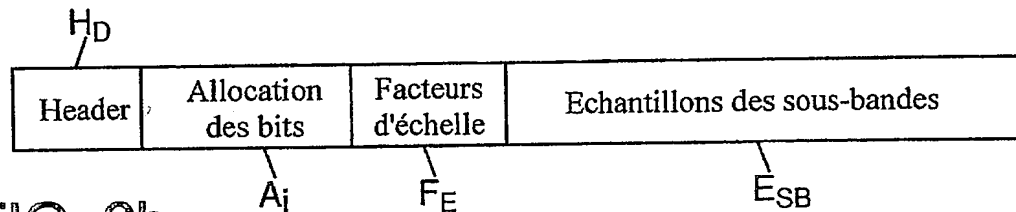


FIG. 6b

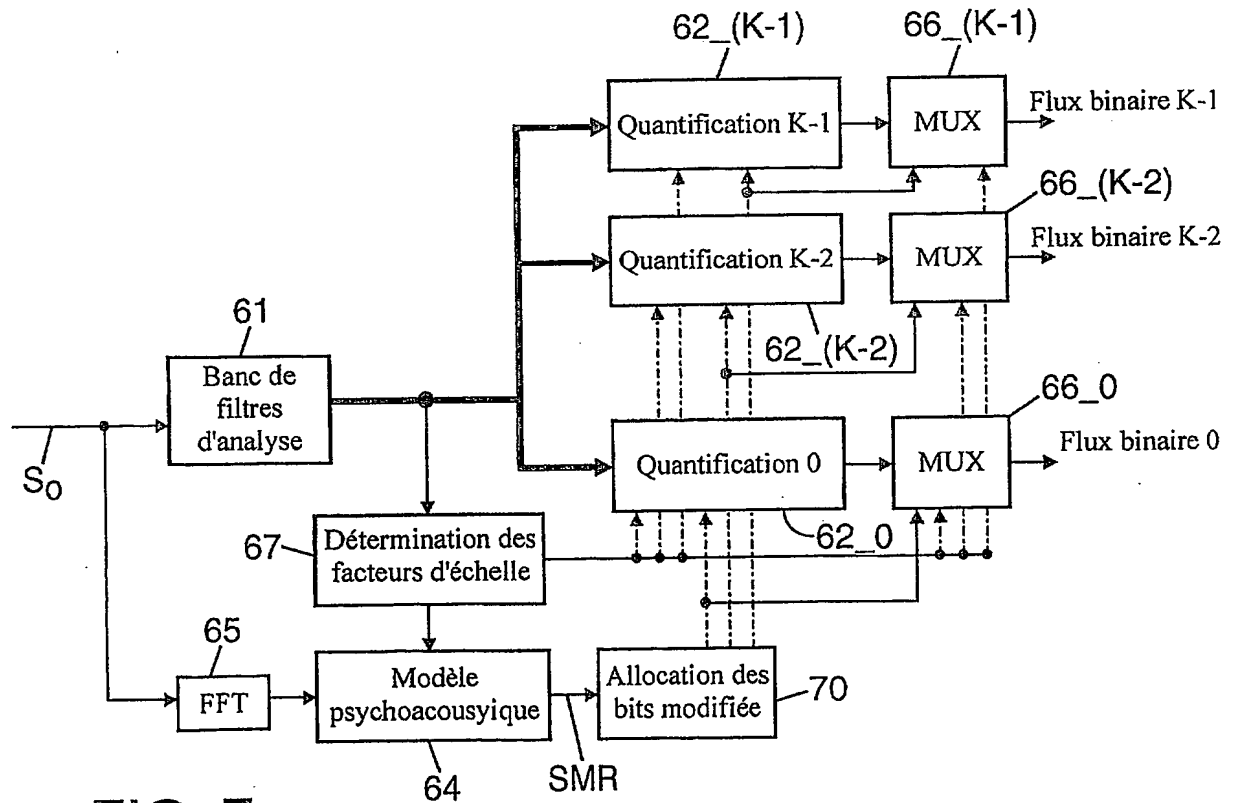


FIG. 7

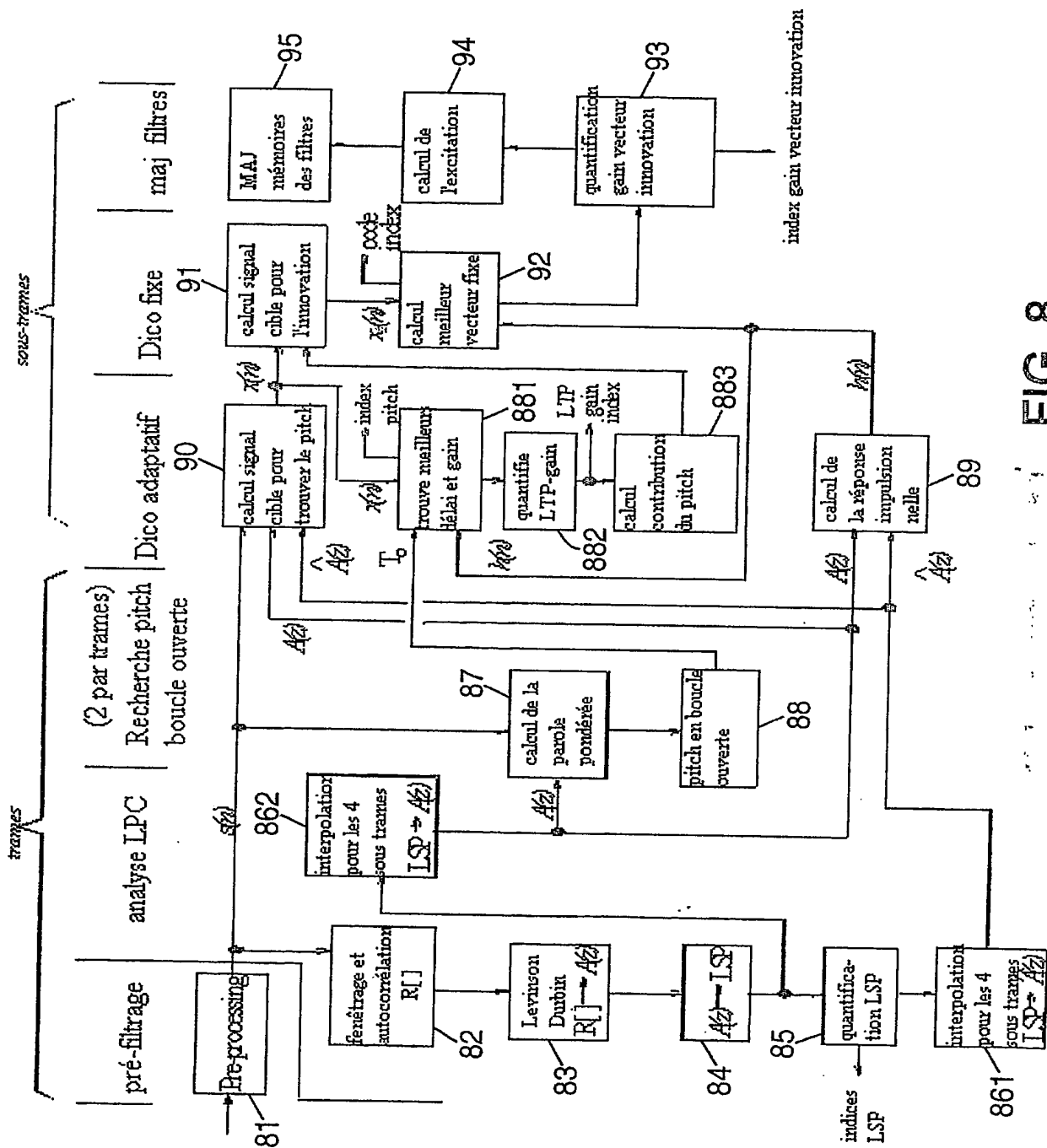
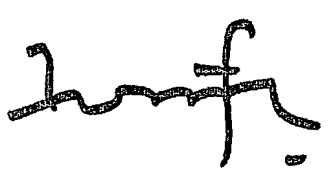


FIG. 8

**DÉSIGNATION D'INVENTEUR(S)** Page N° 1.. / 1..  
(Si le demandeur n'est pas l'inventeur ou l'unique inventeur)

Cet imprimé est à remplir lisiblement à l'encre noire

CB 113 W / 260899

<b>Vos références pour ce dossier</b> (facultatif)		AH/EMA-BFF030328	
<b>N° D'ENREGISTREMENT NATIONAL</b>		0316690	
<b>TITRE DE L'INVENTION</b> (200 caractères ou espaces maximum)			
PROCÉDE DE CODAGE MULTIPLE OPTIMISE.			
<b>LE(S) DEMANDEUR(S) :</b>			
FRANCE TELECOM			
<b>DESIGNE(NT) EN TANT QU'INVENTEUR(S) :</b> (Indiquez en haut à droite «Page N° 1/1» S'il y a plus de trois inventeurs, utilisez un formulaire identique et numérotez chaque page en indiquant le nombre total de pages).			
<b>Nom</b>		VIRETTE	
<b>Prénoms</b>		David	
<b>Adresse</b>	<b>Rue</b>	25, avenue Ernest Renan	
	<b>Code postal et ville</b>	22300	LANNION / FRANCE
<b>Société d'appartenance</b> (facultatif)			
<b>Nom</b>		LAMBLIN	
<b>Prénoms</b>		Claude	
<b>Adresse</b>	<b>Rue</b>	13, rue Ernest Renan	
	<b>Code postal et ville</b>	22700	PERROS GUIREC / FRANCE
<b>Société d'appartenance</b> (facultatif)			
<b>Nom</b>		BENJELLOUN TOUIMI	
<b>Prénoms</b>		Abdellatif	
<b>Adresse</b>	<b>Rue</b>	Résidence Eden Park	
	<b>Code postal et ville</b>	22300	LANNION / FRANCE
<b>Société d'appartenance</b> (facultatif)			
<b>DATE ET SIGNATURE(S)</b> <b>DU (DES) DEMANDEUR(S)</b> <b>OU DU MANDATAIRE</b> (Nom et qualité du signataire) Le 10 décembre 2003  CABINET PLASSERAUD Raphaël LOUISET (CPI 02-1002)			

FR 04 3009





**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record.**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**